Chapter 8

RESULTS AND DISCUSSIONS

8.1 Introduction

The outputs generated by each of the modules were analysed to find the various categories of outputs generated. This analysis helped to modify the rules used by each module and the dictionary content design for better results. The analysis was conducted in three levels: the analyzer level, parser level and the MT system as a whole. The analysis for each phase was carried out using a test corpora which was a collection of naturally occurring text from many domains. The test data was selected from randomly selected news, literature and children stories. Care had been taken to ensure that sentences used a variety of constructs. All possible constructs including simple as well as complex ones were incorporated in the test data set.

8.2 Morphological analyser

The morphological analyzer generated three kinds of output for the compound words in the tested texts.

1. **Correct unique splits**: Here the analyser generated only one split which is the correct split for the compound word.

   ഭംഗിയായി (beautifully) = ഭംഗി + യായി

2. **Correct multiple splits**: Due to ambiguity in the splitting rules in Malayalam language some compound words are split into more than one set of morphemes. All of them are correct according to the syntax rules of the language. More
context knowledge is needed to arrive at the correct translation for the compound word.

The compound word പടക്കത്തിന്റെ has three splits:

a) പട (army) + കത്ത് (letter) + ഇന് (case suffix)
b) പിര (picture) + കത്ത് (letter) + ഇന് (case suffix)
c) പടക്കം (crackers) + ഇന് (case suffix)

3. **Multiple splits with incorrect split**: For some compound words the system generated multiple splits. But the correct one could be identified using the syntax rules of the language.

The compound word കരിമ്പനക്ക് has two splits:

a) കരിം (black) + പന (palm) + ക്ക് (suffix)
b) കരിമ്പ് (sugarcane) + അനക്ക് (move)

The result (b) is syntactically wrong and the parser will be able to identify the correct split from the set of splits. Because according to the syntax rules identified the verb root will have a attached suffix and it never appears in the free form in sentences.

**8.3 Parser**

The parser generated three kinds of output:

1. **Unique correct parse**

In this the parser generated a single parse tree which is syntactically correct.

8.3.1 കേഡിലക്ക് മിക്കവും നാവു
Results and Discussions

(Ravi went to buy book)

Source parse tree:  S(N(നാരി/Ravi) N (നാരി/book) VG(INF(V (ആഴ്ത്ത/buy)
INFA (ഇല്ല/to)) V (ഇല്ല/went)))

Target parser tree:  S(N(Ravi) VG(V(went)INF(INFA(to) V(buy)) N(book))

2. Wrong parse

The correct translation of the following sentence could not be found with our present set of syntax rules and the structure transfer rules.

Malayalam: മഹാനാശാന്ത കെട്ടി കന്നി

Correct translation: Raman heard the story which Mohan told

Source Parse tree:  (S(ADJCNP(ADJC(N(നാരി/Mohan)V (ആഴ്ത്ത/ told)) RP(a/which))N (അമ/story )) N(അമ/Raman) V(അമ/heard))

Target parse tree:  (S( ADJCNP(N (story ) ADJC(RP(which)S(N(Mohan) V(told))))))V(heard) N(Raman)

Output from the translator : Story which Mohan told heard Raman

3. Multiple parses which includes correct parse

An example for this case is given below. The parse trees generated for the two parses are given in Figure. 8.2 and Figure 8.3. In this the correct target parse tree is the second one shown in Figure 8.3.

Malayalam: മോഹൻ കാത്തിൽ സുഗമായി കഴിഞ്ഞിട്ടിട്ടുള്ള (mOhan kaattil sughamaayi kazhinjinju)
Figure 8.1 Multiple source parse trees
Results and Discussions

Figure 8.2 Multiple target parse trees
Two more examples for multiple parse trees are given below. In example 1, b is the correct parse. In example 2 b is the correct parse.

1. (The sea joined the land)
   a. Source parse: \( S(N(\text{sea}/\text{sea}) \ N(\text{land}/\text{land}) \ N(\text{tile}/\text{tile}) \ V(\text{joined}/\text{joined})) \)
      Target parse: \( S(N(\text{sea}/\text{sea}) \ V(\text{joined}/\text{joined}) \ N(\text{land}/\text{land}) \ N(\text{tile}/\text{tile})) \)
   b. Source parse: \( S(N(\text{sea}/\text{sea}) \ NG(N(\text{land}/\text{land}) \ NA(\text{to}/\text{to}) \ V(\text{joined}/\text{joined}))) \)
      Target parse: \( S(N(\text{sea}/\text{sea}) \ V(\text{joined}/\text{joined}) \ NG(NA(\text{to}/\text{to}) \ N(\text{land}/\text{land}))) \)
   c. Source parse: \( S(N(\text{sea}/\text{sea}) \ N(\text{land}/\text{land}) \ VG(V(\text{run}/\text{run}) \ V(\text{joined}/\text{joined}))) \)
      Target parse: \( S(N(\text{sea}/\text{sea}) \ VG(V(\text{run}/\text{run}) \ V(\text{joined}/\text{joined})) \ N(\text{land}/\text{land})) \)

2. The system has been implemented in such a way that the parser finds all parses valid for the current input and the translator creates the translations corresponding to each of these parses. The selection of the correct translation is left to the user. This has been one of the drawbacks of the system which should be rectified using appropriate techniques.

The system has been implemented in such a way that the parser finds all parses valid for the current input and the translator creates the translations corresponding to each of these parses. The selection of the correct translation is left to the user. This has been one of the drawbacks of the system which should be rectified using appropriate techniques.
eliminate wrong tags for a word examining adjacent tags also can reduce the wrong parses greatly. Keeping agglutination minimum in the source language sentences can also reduce wrong parses.

8.4 Prototype Translator

The following three kinds of translations are obtained:

1) Incorrect translation due to syntax limitation: There are many cases where the output is incorrect due to limitations in the syntax. Many sentences do not fall in the structure we have derived. We have assumed that clauses cannot overlap. This creates problems in translation of following type of sentence which is very common in spoken language. One of the cases is shown below:

Malayalam: രാമാനി മഹാന്റെ കഥ ക്ക് കേട്ടി
Parsed output: (S (NG (ADJC (രാമാനി മഹാന്റെ) കേട്ടി)) V (ക്ക്))
Translated output: story which Raman told Mohan heard.
Correct translation: Raman heard the story which Mohan told.

Here the error occurred because രാമാനി കേട്ടി is the principal clause and according to our syntax rule this should come after the subordinate clause മഹാന്റെ (adjective clause) and കേട്ടി, the noun it qualifies. The system will not be able to generate the correct translation for the following word order for the input sentence as it violates the rule for the input that the subject should come in the first place in the sentence:

Malayalam: മഹാന്റെ കഥ രാമാനി കേട്ടി
Translated output: the story which Mohan told heard Raman.
2) **Correct unique translation**: In many cases it generates a single translation which is correct in structure and meaning.

Malayalam: രാമൻ ഒരു കുട്ടിയായി അടിച്ചു

Translated output: Raman hit a boy

3) **Multiple translation with correct translation**: The system generates multiple translations due to two reasons a) the splitter create more than one split for a sentence. b) The parser generates more than one parse tree for a sentence since words can have multiple tags.

a) due to error from the splitter

സിമാഹ്‌ഡി will be split into

സിമാ (money) + ഹ്യാപ (box) or സി (field) + ഹ്യാപ (box)

In the above case there are two ways to split the compound word. This creates multiple translations only one of which will be true in a particular context. More context information is required to find the correct translation.

b) multiple parse tree

Malayalam: മഹാത്മയിൽ അന്തർ പ്രദേശത്ത് കിടക്കും

Morpheme sequence: മഹാ + അന്തർ + പ്രദേശം + കിടക്കും

Translated output: i) Mohan happily lived/finished in forest
  
  ii) Mohan became in forest happy

The word കിടക്കും has two tags. An adverb suffix(ADVA) and a verb(V). The parser creates two parse trees in each of the two cases. കിടക്കും is treated as an adverbal suffix in the first sentence and a verb in the second sentence.
8.5 Sample Outputs from the translator

Sample outputs generated by the translator for the input story given below are shown in Table 8.1. A screenshot from the prototype translator is shown in Figure 8.3.

**Input story in Malayalam:** ഒരു നല്ലവനായ രാജാവ് ഒരിടയത്താരിടത്ത് ഉണ്ടായിരുന്നു. രാജാവിയെ മകള് അതിസുന്ദരി ആയിരുന്നു. ആ രാജകുമാരിക്ക് സൂരയയൊയെ തിളങ്ങുന്ന ഒരു സ്വരണാണ്ടായിരുന്നു. രാജകുമാരി പൂമതാട്ടത്തില് പുണ്ടങ്ങളിക്കാന് ഇഷ്ടമായിരുന്നു. രാജകുമാരി ഒരു ദിവസം കളിച്ചു യകാണ്ടിരുന്നമൊള് സ്വര്ണപത്ത് അടുത്തുള്ള കിണറ്റില് വണ്ടുവണ്ടി. രാജകുമാരി കരയാന്ന ടിങ്ങിരക്കുന്ന ഒരു തവള ഇയതല്ലാം കാണുന്നുണ്ടായിരുന്നു. തവള രാജകുമാരിയുയട അടുമത്തക്ക് ചാടിയെന്നുത് തവള പത് എടുത്തുതരാം എന്ന് തവള രാജകുമാരിമയാട് പറഞ്ഞു. രാജകുമാരി നൂറഞ്ച് സ്വരണനാണയങ്ങള് യകാണ്ട് എന്ന് തവളമയാട് പറഞ്ഞു. എനിക്ക് രാജകുമാരിയുയട കൂയട എണ്ണം ഇരിക്കണം. രാജകുമാരിക്കു തവളയുയട ആവശ്യം ഇഷ്ടയെട്ടില്ല. രാജകുമാരി എങ്കിലം തവളയുയടനടക്കാന് സമ്മതിച്ചു. യപ്പാട്ടു തവള കിണറ്റിമെക്ക് ചാടി. തവള പത് യകാടുത്തമൊള് രാജകുമാരിയുയട ഭാവം മാറഞ്ഞു. രാജകുമാരി തവളയല്ലാ യകാണ്ടുമപാകാന് തയ്യാറായില്ല. രാജകുമാരി രാത്രിയില് ആഹാരം കഴിക്കാന് ഇരുന്നമൊള് തട്ടി. രാജകുമാരി അച്ഛമനാട് തവളയല്ലാ പറഞ്ഞു. അവള് തവളയുയട ഇരിക്കണം. രാജാവ് രാജകുമാരിമയാട് പറഞ്ഞു. രാജകുമാരി തവളയായ അകത്ത് വരാന് സമ്മതിച്ചു. അങ്ങനെ തവള രാജാവിയും രാജകുമാരിയുയടയും കൂയടയിരുന്ന് കഴിച്ചു. രാജകുമാരി രാത്രിയില് ഉറഞ്ഞാന് കിടന്നമൊള് തവള കൂയടകിടന്നു. രാജകുമാരി എയന്ന പനയക്ക് തടവണം. തവള രാജകുമാരിമയാട് പറഞ്ഞു.
### Table 8.1 Sample outputs from the prototype translator

<table>
<thead>
<tr>
<th>System Input and Output</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Input</strong>: a kind king in a place was.</td>
<td>System output is in line with the English version except for the positioning of the article. Meaningfully correct sentence.</td>
</tr>
<tr>
<td><strong>English version</strong>: There was a kind king in a place.</td>
<td></td>
</tr>
<tr>
<td><strong>System Output</strong>: A kind king was in a place.</td>
<td></td>
</tr>
<tr>
<td><strong>2. Input</strong>: King’s daughter very beautiful was</td>
<td>System output is exactly in line with the English version except for the positioning of article to the noun king since same word is not there in input language. Meaningful correct translation.</td>
</tr>
<tr>
<td><strong>English version</strong>: The king’s daughter was very beautiful.</td>
<td></td>
</tr>
<tr>
<td><strong>System Output</strong>: King’s daughter was very beautiful.</td>
<td></td>
</tr>
<tr>
<td><strong>3. Input</strong>: That princess sun like shining is a goldenball.</td>
<td>System output gives translation without positioning of article for the nouns. The variations in translations for am have not been considered. System output is meaningful correct translation.</td>
</tr>
<tr>
<td><strong>English version</strong>: The princess had a golden ball which was shining like the sun.</td>
<td></td>
</tr>
<tr>
<td><strong>System Output</strong>: That princess had a ball which is shining like sun.</td>
<td></td>
</tr>
<tr>
<td><strong>4. Input</strong>: Princess garden in ball play to liked</td>
<td>Output gives translation without positioning of preposition since same word is not there in input language Meaningful correct translation.</td>
</tr>
<tr>
<td><strong>English version</strong>: The princess liked to play in the garden.</td>
<td></td>
</tr>
<tr>
<td><strong>System Output</strong>: Princess liked to play in garden.</td>
<td></td>
</tr>
<tr>
<td><strong>5. Input</strong>: princess a day play was when golden ball nearby well in fell.</td>
<td>Output translation is without positioning of preposition since same word is not there in input language. Multiple translations of am has not been considered. Meaningful correct translation.</td>
</tr>
<tr>
<td><strong>English version</strong>: When the princess was playing one day the golden ball fell into a nearby well.</td>
<td></td>
</tr>
<tr>
<td><strong>System Output</strong>: When princess was playing a day golden ball fell in nearby well.</td>
<td></td>
</tr>
<tr>
<td><strong>6. Input</strong>: Princess cry to started</td>
<td>System output gives exact English version of input sentence.</td>
</tr>
<tr>
<td><strong>English version</strong>: The princess started to cry.</td>
<td></td>
</tr>
<tr>
<td><strong>System Output</strong>: Princess started to cry.</td>
<td></td>
</tr>
<tr>
<td><strong>7. Input</strong>: well side in sat a frog all these see was</td>
<td>Meaningful correct translation which slightly varies from the English version.</td>
</tr>
<tr>
<td><strong>English version</strong>: A frog sitting beside the well was seeing all these.</td>
<td></td>
</tr>
<tr>
<td><strong>System Output</strong>: frog which was sitting in side of well was seeing all these.</td>
<td></td>
</tr>
<tr>
<td><strong>8. Input</strong>: frog princess’s side jumped went</td>
<td>Meaningful correct translation which slightly varies from the English version.</td>
</tr>
<tr>
<td><strong>English version</strong>: The frog jumped to the princess.</td>
<td></td>
</tr>
<tr>
<td><strong>System Output</strong>: frog went jumping to princess’s side.</td>
<td></td>
</tr>
</tbody>
</table>
Results and Discussions

Figure 8.3 Screenshot from translator
8.6 Merits of the prototype translator

1. The translator could handle both simple and complex sentences. Correct translations for complex sentences with two adverb or noun clauses could be generated.

2. The model adopted for the parser makes it possible to extend the parser to handle words of any depth by enhancing the syntax rules.

3. The system design was based on artificial intelligence techniques. The parser developed was a general one and the same could parse sentences in any language if the syntax rule were replaced with that of the new language.

4. The translation systems for other language pairs could easily be generated by replacing the syntax rules and structure transfer rules with those of the new language pair.

5. The modular design helps to reuse the modules for other applications. The analyzer and parser can be integrated into the present day Malayalam spell checkers to enhance their performance. They also can be integrated into Malayalam language understanding systems.

6. All the modules work with morphemes. This helps to reduce the space required by the dictionary.

8.7 Limitations of the MT system

The translator could not find the correct translations of sentences which are not of the syntax rules selected as give in an example shown above. In addition to this the following limitations were identified in the outputs generated by it.
1) Limitations in agreements and surface form generation

Agreements like subject verb agreements or morphological generation for surface forms for plural etc. are missing.

Malayalam: കുട്ടികള് ക്കുന്നു

Translated output: Child(s) is/are playing

The system was unable to find the correct translation “are playing” and also the correct plural form for the word child as children.

2) Limitations in word sense disambiguation

The parser could perform word sense disambiguation only based on morpheme tags. So the system could not find the correct translation for polysemous words. Multiple meanings for the morpheme category are produced.

i) The system could not distinguish between the two translations for the locative case suffix ഇല് in the following sentences

a) Malayalam: രാമ തറായി ഇരുന്നു (rama thaRayil irunnu)

Translated output: Rama sat on/to the floor

b) Malayalam: രാജു ഇല് മപായി (raaju delhiyil pOyi)

Translated output: Raju went on/to Delhi

ii) In the following set of sentences the word ആയി has to be treated differently which is not done by the system. In sentence a) the ആയി suffix should be translated as ‘as’ whereas in sentence b) it should be translated as the suffix –ly
which should be attached to the morpheme beautiful forming the adverb beautifully.

a) Malayalam: രാമു മമാഹന് കുറഞ്ഞ് പണം കടമായി നല്കി

Translated output: Ramu gave some money as/ly loan to Mohan

b) Malayalam: രാമു പടം ഭംഗിയായി വരച്ചു

Translated output: Ramu drew beautiful as/ly the picture

3) Absence of a transliteration module and named entity recognizer

The system does not possess a named entity recognizer and a transliteration module. Their addition would make handling of out of vocabulary words in an efficient way.

8.8 Conclusion

This chapter discussed the analysis of the outputs generated by the system to find the various categories of outputs generated. This analysis helped to modify the rules used by each module and the dictionary content design for improving the performance of the system. The analysis was conducted in three levels: the analyzer level, parser level and the MT system as a whole. Even for sentences with more than two subordinate clauses the system returned meaningful translations. All the modules generated multiple outputs which reduces the accuracy of the system. The prototype system also produced multiple translation results for some inputs. They were due to the deficiency in the analyzer module or the parser module. Only word sense disambiguation based on lexical category of the words in the sentence could be performed by the parser. The system also produced error outputs in some cases due to the
difference in syntax. The system also lacks in considering the agreements between chunks in the sentence. More rules can be added to make the system to give exact translation of input sentences in all cases. Additional modules like finding and replacing collocations, finding and replacing named entities can also be added to the basic translator. The next chapter discusses the performance evaluation of the prototype system.
Chapter 9

PERFORMANCE EVALUATION

9.1 Introduction

This chapter discusses the evaluation results of the developed prototype system. First the storage space requirement for the dictionary is presented to show the saving in storage space achieved by storing morphemes other than the inflected forms for words. The evaluation of each module and also the prototype as a whole are discussed. Results of the comparative study of the performance of other existing systems which were developed for Indian language translation are also discussed.

9.2 Analysis of space requirement for dictionary

The space efficiency achieved by a morpheme based dictionary can be shown as follows. Only noun and verb inflections are considered for the present calculation. Also it is assumed that a noun or a verb can have only one suffix. In fact Malayalam is highly agglutinative and any number of morphemes can join to form a word. In those cases the space saving achieved will be much higher compared to the one shown here.

- Number of nouns in the dictionary - N
- Number of verbs in the dictionary - V
- Total space required for the nouns - Sn bytes
- Total space required for the verbs - Sv bytes
- Total space required by noun suffixes - Ns bytes
- Total space required by verb suffixes - Nv bytes
- Total number of noun suffixes - n
- Total number of verb suffixes - m
Space Saving  = Space required with inflected words - Space required with morphemes  
\((Sn*n+Sv*m+Ns*N+Nv*V) - (Sn+Sv+Ns+Nv)\)

The sample dictionary used consists of:

- Total 1500 words
- N 775
- V 128
- Sn 13298 bytes
- Sv 2735 bytes
- Ns 34 bytes
- Nv 96 bytes
- n 17 (case and number suffixes)
- m 12 (Tense, aspect and mood suffixes)

Saving in storage space  =\((13298*17+2735*12+34*775+96*128) - (13298+2735+34+96)\) = 281361 bytes

I.e. saving% = 94.5%

In this study it was assumed that one verb or one noun can have only one suffix at a time. But in actual case in Malayalam any number of morphemes can join to form a word. In that case the saving in space will be much higher.

### 9.3 Test data set

MT evaluations typically include evaluation of the quality of the unedited translations, which includes intelligibility, accuracy, fidelity, appropriateness of style, the usability of facilities for creating and updating dictionaries, the
extendibility to new language pairs and/or new subject domains and cost-benefit comparisons with human translation performance. The performance evaluation of the prototype translation system was conducted during various stages of its development. The performance of each of the phases was separately carried out. The initial evaluation was carried out using test suites which was a collection of artificially constructed inputs, where each input was designed to probe a system's treatment of a specific phenomenon or set of phenomena. Inputs were in the form of sentences, sentence fragments, or even sequences of sentences. The feedback from the tests was used to modify the system for better performance. The final evaluation for each phase was carried out using a test corpora which was a collection of naturally occurring text from many domains. The test data was selected from randomly selected news (articles sports, politics, world, regional, entertainment, travel etc., literature text and children stories. Care had been taken to ensure that sentences used a variety of constructs. All possible constructs including simple as well as complex ones were incorporated in the set. The sentence set also contained all types of sentences such as declarative, interrogative, imperative and exclamatory. Sentence length was not restricted although care had been taken that single sentences were not too long. The test data set is shown in Table 9.1.

Table 9.1 Test data set for evaluation of prototype system modules and whole MT system

<table>
<thead>
<tr>
<th></th>
<th>Children stories</th>
<th>News articles</th>
<th>Literature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of documents</td>
<td>30</td>
<td>20</td>
<td>10</td>
</tr>
<tr>
<td>Number of sentences</td>
<td>1542</td>
<td>423</td>
<td>2021</td>
</tr>
<tr>
<td>Number of words</td>
<td>15431</td>
<td>2811</td>
<td>14356</td>
</tr>
</tbody>
</table>
The performance evaluation of the system was conducted in three levels: In the analyzer level, parser level and the MT system as a whole. Out of the different metrics for performance evaluation we had chosen accuracy test and error analysis for the MT system and the different modules. Accuracy Test was conducted to measure how much information the translated sentence retained compared to the original. In error analysis mainly word error rate was calculated.

9.4 Performance evaluation of morphological analyser

The outputs from the morphological analysis was analysed and the accuracy of the module was calculated. The performance of the analyser for the three kinds of texts is shown in Table 9.2 and Figure 9.1. The compound words which fall into the first category of correct unique split and the number of constituents in a compound word were found to be more in news articles than children stories. The analyzer is found to be working with 90.7% accuracy.
Table 9.2 Performance of morphological analyser

<table>
<thead>
<tr>
<th>Type of output</th>
<th>Children stories</th>
<th>News articles</th>
<th>Literature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correct unique split</td>
<td>90%</td>
<td>94%</td>
<td>88%</td>
</tr>
<tr>
<td>Correct multiple split</td>
<td>8%</td>
<td>4%</td>
<td>8%</td>
</tr>
<tr>
<td>Multiple splits with incorrect split</td>
<td>2%</td>
<td>2%</td>
<td>4%</td>
</tr>
</tbody>
</table>

Figure 9.1: Performance of the morphological analyzer
9.5 Performance evaluation of parser module

The parser was tested with test suits initially for arriving at the correct syntax of sentences and also for identifying the correct chunks for reordering. Finally the module was tested with a test corpus from three domains children stories, news articles and literature texts. Two kinds of evaluation were done with parser output. Based on the type of output sentences produced and based on the chunks it generated. The precision and recall percentages were used for evaluation based on chunks.

9.5.1 Performance evaluation on a sentence level

The results are tabulated in Table 9.3 and shown in Figure 9.2. The analysis of the performance showed that the parser works well for children stories as the syntax rules derived closely matches with that of children stories.

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Children stories</th>
<th>News</th>
<th>Literature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Unique correct parse</td>
<td>85</td>
<td>78</td>
</tr>
<tr>
<td>2.</td>
<td>Wrong parse</td>
<td>5</td>
<td>12</td>
</tr>
<tr>
<td>3.</td>
<td>Multiple parses which includes correct parse</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>
The precision for the parser was based on the kind of chunks it produced. Only the F-measures for noun chunks, verb chunks and clause chunks are shown in Table 9.4. Similar calculations were done for other kinds of chunks. Average precision for the chunking process was found to be 91.2% and the recall was found to be 89.3%.

\[
\text{Precision} = \frac{\text{Number of correct chunks in output}}{\text{Total number of generated chunks}}
\]

\[
\text{Recall} = \frac{\text{Number of correct chunks in output}}{\text{Total number of chunks in the correct translation}}
\]

The precision and recall were used to calculate the F-measure.
\[ F = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \]

**Table 9.4 Precision and Recall from the parser module**

<table>
<thead>
<tr>
<th>Chunk type</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noun chunk</td>
<td>95%</td>
<td>96%</td>
<td>95.4</td>
</tr>
<tr>
<td>Verb chunk</td>
<td>93%</td>
<td>94%</td>
<td>93.49</td>
</tr>
<tr>
<td>Clauses</td>
<td>90%</td>
<td>93%</td>
<td>91.47</td>
</tr>
</tbody>
</table>

### 9.6 Performance evaluation of the prototype MT system

Out of the different metrics for performance evaluation we had chosen accuracy test and error analysis for the MT system. Accuracy Test was conducted to measure how much information the translated sentence retained compared to the original. In error analysis mainly word error rate was calculated. A study was also conducted to compare the performance of our system with that of four other translation systems for Indian languages.

#### 9.6.1 Accuracy testing

The performance of the prototype system for various types of documents is shown in Table 9.5 and Figure 9.3. A careful analysis of the table shows that the system is most suited for translation of children stories. This is because word compounding is less in children stories and the syntax structure closely
matches with that of children stories. The system gives an average accuracy rate of 65%.

Table 9.5 Performance of prototype system

<table>
<thead>
<tr>
<th>Type of output</th>
<th>Children stories</th>
<th>News articles</th>
<th>Literature Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Incorrect translation due to syntax limitation</td>
<td>25%</td>
<td>50%</td>
<td>30%</td>
</tr>
<tr>
<td>2. Correct unique translation</td>
<td>55%</td>
<td>30%</td>
<td>40%</td>
</tr>
<tr>
<td>3. Multiple translation with correct translation</td>
<td>20%</td>
<td>20%</td>
<td>30%</td>
</tr>
<tr>
<td>a) multiple parses</td>
<td>8%</td>
<td>5%</td>
<td>15%</td>
</tr>
<tr>
<td>b) limitations in word sense disambiguation</td>
<td>8%</td>
<td>15%</td>
<td>15%</td>
</tr>
</tbody>
</table>

Figure. 9.3 Graph depicting performance of prototype MT system
9.6.2 Comparison with existing systems

The performance of the system had been compared with that of some of the existing MT systems for Indian languages which use the same approach as that of the developed prototype system. Three MT systems had been taken for the comparative study. The results are tabulated in Table 9.6.

<table>
<thead>
<tr>
<th>MT system</th>
<th>Structure Transfer approach</th>
<th>Sentence Types</th>
<th>Accuracy of chunking</th>
<th>Accuracy of translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mantra</td>
<td>LTAG parser</td>
<td>Office administration</td>
<td>Not available</td>
<td>93%</td>
</tr>
<tr>
<td>Punjabi-English</td>
<td>Hand crafted chunk rules</td>
<td>Simple sentences in Legal documents</td>
<td>86.57%</td>
<td>70.54%</td>
</tr>
<tr>
<td>English-Malayalam</td>
<td>Stanford parser</td>
<td>Simple sentences with max. of 6 words</td>
<td>Not available</td>
<td>53.63%</td>
</tr>
<tr>
<td>Malayalam-English</td>
<td>Top down parser</td>
<td>Complex sentences, Children stories</td>
<td>85%</td>
<td>75%</td>
</tr>
</tbody>
</table>

A detailed Comparison of the prototype system developed with Punjabi to English machine translation system had been conducted since both use transfer
based approach as the basic approach for translation. Both the systems use parsing, chunking and structure transfer for the generation of translated sentences[116].

The English to Punjabi translation system developed by Punjabi University [116] is based on the transfer approach, with three main components: an analyzer, a transfer component, and a generation component. The analysis component is composed of preprocessing, morph analyzer and tagger, phrase chunker and the module which adds semantic information, assigns morph information and karaka roles to the phrases by means of Punjabi grammatical rules. The target language translation is produced by means of a combination of synthesis and post processing modules. It also includes a transliteration module. The morph database used for the system includes words in Punjabi along with the information about its gender, number, person, case, tense etc. Every inflected word also contains the root word from where it is derived. The database also contains the grammatical category of each word and also the inflected words it can form. Only noun phrases, adjective phrases, postpositional phrases and verb phrases have been considered in the system. The system can handle only simple sentences. Reported translation accuracy is 60.33%, Tagging accuracy is 75.54% and chunker accuracy is 86.57%.

The Malayalam to English translation system developed contains only base morphemes, the lexical category and the translation. The system uses only one bilingual dictionary which is shared by morphological analyser, parser and generation component. The dictionary presently contains only a small set of commonly found words in Malayalam. The dictionary size required is small compared to the former one as it contains only the root words. The system uses the top down parser to do word sense disambiguation based on lexical category,
chunking and structure transfer. The system can handle simple and complex sentences with any number of adjective clauses. The system can recognise more chunks compared to the previous one. A transliteration module is absent in the present system and it is assumed that all words including proper nouns are present in the dictionary. The present prototype system takes care of complex sentences with utmost two adverb clauses or noun clauses. Chunking accuracy of 93% has been obtained and the translation accuracy including simple and complex sentences was found to be 75%.

9.6.3 Analysis of word Error rate

The Word error rate for different types of input is shown in Figure 9.4. It is found that word error rate is more for news articles compared to other types of documents. After robust analysis, Word Error rate was found to be 8% which was comparably lower than that of general systems, where it ranges from 9.5 to 12. Figure 9.5 shows the percentage type of errors out of the word errors found. The addition or omission of words like omission of articles was the most frequently occurring word error.
Figure 9.4 Word error rate for various texts

Figure 9.5 Word error rate
9.7 Conclusion

The chapter discussed the performance evaluation of the prototype translation system. The performance of the morphological analyzer, the parser and the prototype system as a whole were evaluated separately using texts belonging to different domains. The MT system was evaluated based on accuracy rate, word error rate. The average accuracy rate for the system was found to be 65% and 75% for children stories. The results of the comparison of performance of our prototype with other existing translation systems also were discussed in the chapter.