Error Patterns and Analysis of Hindi Shallow Parser

Prabhas Tiwari
Jamia Hamdard University
Hamdard Nagar
New Delhi, India

Md. Tabrez Nafis
Jamia Hamdard University
Hamdard Nagar
New Delhi, India

ABSTRACT
Simplification is an integral part of Machine translation, however, it still remains the most complex part of the process. A sentence in Hindi can be written in multiple ways. They can be complex sentences or simple ones. These sentences then need to be translated into English language. For this, the complex sentences need to be converted into simple sentences before being translated.
This paper concerns Sentence Simplification of Complex Hindi Sentences for the Hindi shallow parser that was developed by IIIT. Complexity of a sentence can occur due to presence of clauses, multiple verbs and usage of conjunctions. So splitting the sentence works if the above mentioned properties of the sentence can be eliminated. To automate the process, a simplification algorithm has been formed. The paper shall talk about errors and patterns that have been analyzed, so as to improve the accuracy of the simplified sentence, and preserving the sense of the original sentence.

General Terms
Computational Linguistics, Natural Language Processing, Sentence Simplification, Shallow Parser, Translation.

1. INTRODUCTION
It has been found out in many psychological studies that the comprehension of a given statement is easier if the statement is simple[1]. The moment we start adding complexities, could be lexical, or syntactical, the understanding of the statement reduces and it takes more effort to find out the sense or the meaning behind the statement. Before moving ahead on this, we need to consider a few properties and differences in the Hindi language, as the language rules of Hindi and English languages are not the same.

Ram finished his work.
राम ने अपना कार्य समाप्त कर लिया |
We notice that, in English, it is easier to identify the proper noun, as it starts with capital letters. But in Hindi language, no such distinction can be made, as there is no existing concept of upper or lower case in the Devanagari script.

He fell down while he was running.
वह दौड़ते-दौड़ते गिर गया|

Here, the verb "to run" has been used in the continuous tense form. While translating it to Hindi language, we observe a verb repetition "दौड़ते-दौड़ते".

Further, there can never be an absolute translation from one language to another. There might exist a vocabulary related complexity (lexical) or a difference in phrase formation, depending upon the grammar of the language, and usage by the speaker, whether native or not (syntactical complexity).

Due to such divergences in languages, Machine Translation has always been a tough task. Talking about the Hindi-English translation, noting the above divergences, accuracy is not very high in the existing system. To improve accuracies, it is, thus, highly important that the phrase that is being sent as an input to the translator is kept in reduced simple form.

Reducing a sentence in a simple form, is again a tough task. It involves removing assets that add complexity to the sentences. Consider the following sentences in Hindi language. The first sentence is complex, and it is followed by its reduced form:

मोहन खाना खाने के बाद विश्राम करने गया |
[*Mohan rested after eating*]
मोहन ने खाना खाया| उसके बाद मोहन विश्राम करने गया |
[*Mohan ate. Mohan went to rest.*]

उनका कहना था, "हम पीछे नहीं हटेंगे !"
[*He said, "we will not back down!"*]
उन्होने कहा| वह पीछे नहीं हटेंगे|
[*He spoke. He will not back down!*]

श्याम ने एक पीला बड़ा बंगला खरीदा जिसमे एक आंगन भी है|
[*Shyam bought a big yellow house that has a garden as well*]
श्याम ने एक बंगला खरीदा | बंगला पीला और बड़ा है | उसमे एक आंगन भी है |
[*Shyam bought a house. House is big and yellow. It has a garden as well.*]
सीता रोई और फिर सोयी |
[Sita slept after crying.]
सीता रोई | सीता सोयी |
[Sita cried. Sita slept.]

These were a few examples of complex to simple sentence conversions.

2. RELATED WORK
A Hindi language shallow parser has been developed by International Institute of Information Technology, Hyderabad. It has been made available [2]

2.1 Complexity in Sentence
According to Yamuna Kachru (Kachru, 2006), the complexity in a sentence arises when there is a usage of clauses in a sentence, using connectives. When the length of the sentence increases, its complexity too increases (Chandrashekhar et al., 1996). Soni et al. (2013) has also mentioned that the number of verb chunks increases with the length of sentence.

Thus, compiling these sources, we can call a sentence complex, on the basis of the following criteria:

1. Criterion 1: Length of the sentence is greater than 5.
2. Criterion 2: Number of verb chunks in the sentence is more than 1.
3. Criterion 3: Number of conjuncts in the sentence is greater than 0.

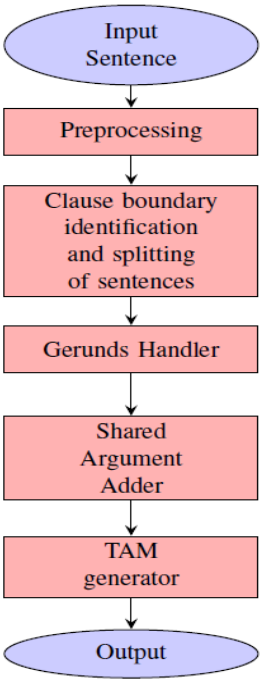
Consider Table 1 for the classification criterions for calling the sentence simple or complex.[4]

Table 1: Criterions for Classification of a Sentence as Simple or Complex

Criterion 1	Criterion 2	Criterion 3	Category of Sentence
No	No	No	Simple
No	No	Yes	Simple
No	Yes	No	Simple
No	Yes	Yes	Simple
Yes	No	No	Simple
Yes	No	Yes	Complex
Yes	Yes	No	Complex
Yes	Yes	Yes	Complex

2.2 Flowchart of Process of Simplification

The following flowchart represent the processing done on the given sentence to convert it into a simple sentence, from its complex being [3]



Our system comprises of a pipeline incorporating various modules. The first module determines the boundaries of clauses (clause identification) and splits the sentence on the basis of those boundaries.

3. PROBLEM STATEMENT

While the given algorithm works fine in simplification process, there still remains a scope of improvement in its accuracy. Many times the parser, while reducing the sentence into its simplest form, could not preserve the sense and the meaning behind the sentence. Though, one thing was proved that, Machine Translation improved if the input sentences in Hindi were first simplified, and then processed for translation into English language.

Thus, being an encouragement, it is indeed important to understand the errors that occurred in the existing system so that they could be removed and the system be improved. This in turn shall be helpful in machine translation tasks of Hindi to English translations.

Like any system, there are a few flaws in the existing system of Sentence Simplification too. To improve this accuracy, we must find out in which all scenarios the algorithm is not giving the expected results. Further, we have analyzed simplification techniques which are best suited so as to preserve the meaning of the sentence, as much as possible

4. SOLUTION

The process of improving any system starts with finding its errors. This paper deals with the Error Patterns, and Analysis; and Suitable Breaking of Sentences for Preservation of Sense of the Sentence

4.1 Error Patterns and Analysis

The following patterns have been identified taking two assumptions solely: errors that are miniscule in terms of verb conversion shall be handled by post processors or word generators. Secondly, sentences with कर have to be ignored for the time being, as no perfect solution can be thought of at the moment, whether to take it as an acting verb or a supporting verb.

We have been able to identify 5 patterns. These patterns occurred in 35 sentences for one or more times out of a total of 198 sentence outputs.

This equals to an error of 17.67%.

1. Whenever there a use of a past perfect tense verb has been used, it must not be converted into another verb form.
   The sentence broken at that verb must not be changed, and only converted finite. लौटे,भागे , दौड़े.
   These verbs shall only be proceeded by हैं. लौटे हैं,हैं भागे , दौड़े हैं.

   However in this condition, we need to know the quantity and the form of the noun. If the actor is in plural, OR if there is an actor to whom respect is being imparted, use of हैं)hain(is to be made. In other conditions, only है (hai) shall be used.

2. The parser is making an error whenever a verb is followed by its temporal/spatial/reason or any other property. In these conditions the parser is adding a present tense of "be" rather than breaking the sentence at the VGNN identified. लौटने के बादबाद के उठने ,बाद के जाने ,, करने की.

   These verb forms describe some property of the action being performed. They must be converted to their present form only.

   लौटेक ,उठे ,गए ,ी.

   We see that whatever is followed after the verb is to be removed. Instead, what the parser is doing is adding the words "ना है"

The temporal property must be tagged with the VGF, with an addition of the word, generated by the word generator "इसके बाद", "इस वजह से" etc.

3. When the verb is conjoined with 'ना' the parser is making an error of separating the verb and adding an 'ए'.

   मानना.सोचना ,करना ,कहना ,

   These verbs must not be separated, and kept as they are, with the respected tense form of "be".

   मानना थामा ,है मानना ,नेगा.

   कहना था.कहेगा ,है कहना ,

4. The word तथा is not being identified as the conjunction between two sentences.

5. The verb form of होना is being converted to हुया .This needs to be corrected in the parser's word generator and post processors.

4.2 Sense Preservation based on position of verb breakage

The following are the stats for the sentence breaking in 2 or more VGNN's.

1. When we break the complex sentence at both the VGNN's, the sentence mostly loses its entire meaning. Somewhere along the simplification, the actual sense of the sentence was lost. The correctness in terms of preservation of the meaning of the sentence in this case was a mere 11%.

2. On breaking the sentence at the first VGNN position, we were able to retain some part of the meaning of the sentence. This equalled up to a higher level of correctness, i.e. 31%.

3. On breaking the sentence at the third form, i.e. breaking it at the second VGNN, we received the best meaning suited for the entire sentence. This leads to a correctness level of 45.71%.

4. However, there were many such sentences in the given test cases that contained single VGNN forms, or were in a format that either could not be simplified further or had a usage of कर in the sentence. Such sentences were a total of 12.28% of the entire given test cases.

All the above results were deduced according to this pattern:

When the verb is reflecting on an ongoing action, the sentence must not be broken there. The action must be allowed to complete. In these cases, VGF should not be separated.

This pattern was visible mostly on the second VGNN when the sentence starts describing the ongoing action by the actor on the subject.

5. ACKNOWLEDGMENTS

We would like to thank International Institute of Information Technology, Hyderabad, for providing a platform to research on their shallow parser.

6. CONCLUSION

Therefore the quality and meaning of the sentence was preserved whenever the sentence was broken down at the second position VGNN which is the third case in most of the sentences, where the verb is allowed to complete the action.

Also, the errors were analysed where the simplification parser was not able to return expected results. A list of those scenarios were presented. Working on these errors will significantly improve the accuracy of the shallow parser.

In future, with the help of sentence simplification techniques, the process of sentence translation will become very easy. Currently many organisations use the help of users for translation process. However, this takes a lot of time and energy. If the entire process is automated, a great deal of time can be saved.

Sentence preservation techniques will help preserve the sense of original sentence that requires to be translated. Many a times, when automatic translation techniques are used, the actual meaning of the sentence is lost. Hence, proper understanding of sentence formation and using machine learning techniques will deal with this problem.

7. REFERENCES

[1] Pg. 2, Exploring the effects of Sentence Simplification on Hindi to English Machine Translation System.

[2] ltrc.iiit.ac.in/showfile.php?filename=downloads/shallow_parser.php

[3] Pg. 3, Exploring the effects of Sentence Simplification on Hindi to English Machine Translation System.

[4] Pg. 4, Exploring the effects of Sentence Simplification on Hindi to English Machine Translation System

[5] Chapter 9: Complex and Compound Sentences, from book, Hindi, by Yamuna Kachru.

[6] Guidelines For POS And Chunk Annotation For Indian Languages by Akshar Bharati, Dipti Misra Sharma, Lakshmi Bai, Rajeev Sangal, from Language Technologies Research Centre, IIIT, Hyderabad.

[7] Tree Banks for Indian Languages, Guidelines for Annotating Hindi TreeBank (version – 2.0), by Akshara Barati, Dipti Misra Sharma, Samar Husain, Lakshmi Bai, Rafiya Begam, Rajeev Sangal, Language Technologies Research Center, IIIT, Hyderabad, India.

[8] Exploring Verb Frames for Sentence Simplification in Hindi, Ankush Soni, Sambhav Jain, Dipti Misra Sharma, Language Technologies Research Centre IIIT Hyderabad

[9] Automatic Sentence Simplification for Subtitling in Dutch and English, Walter Daelemans and Anja Hothker and Erik Tjong Kim Sang

[10] A Review of English to Indian Language Translator, Kanika Ankur, Divyanjali, Shalini Mittal