
Indian Institute of Technology, Bombay

Project Report on

# Semi-automated Paraphrase Dataset Creation for Hindi and Malayalam

SUBMITTED TOWARDS THE
PARTIAL FULFILLMENT OF THE REQUIREMENTS OF

**CS626: Speech and Natural Language Processing and the Web**

BY

Shubham Nemani 203050011
Pooja Verma 203050072
Anish M M 203050066

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

# Contents

# 1 Introduction

A paraphrase is a restatement of a text expressed using other words. Alternatively, two texts say text1 and text2 can be defined as paraphrases if they textually entail each other bi-directionally, i.e. if text1 entails text2 and text2 entails text1.

In broader context, paraphrase detection has various applications.One of the most common applications of paraphrasing is the automatic generation of query variants for submission to information retrieval systems.

The aim is to create dataset semi-automatically for paraphrase detection in Hindi and Malayalam language.

# 2 Problem Statement

To the best of our knowledge, the only paraphrase dataset available for Hindi and Malayalam is the DPIL-FIRE 2016 Paraphrase dataset which contains 2500 pairs of sentences for training and 900 pairs for testing for both languages. This size is much smaller compared to paraphrase datasets available for languages like English (like the PAWS-Wiki [1]).

Filling this gap was the motivation for our course project. We aimed to provide a semi-automated process for creating such a dataset for Hindi and Malayalam and provide (1) a large automatically created, unedited dataset with noisy labels, (2) a relatively smaller subset of the above which is manually edited/checked and has labels with no noise, (3) the code for the automated candidate pairs generation. Similar datasets[2] are available for English where a large set of noisily labeled pairs augment a smaller correctly labeled dataset.

# 3 Input Datasets

- IIT Bombay English-Hindi Parallel Corpus:
  The IIT Bombay English-Hindi corpus contains parallel corpus for English-Hindi as well as monolingual Hindi corpus collected from a variety of existing sources.
- Mann ki baat Corpus:
  The Prime Minister's speeches - Mann Ki Baat, on All India Radio, translated into many languages.

# 4 Methods

## 4.1 Translation from parallel corpora

We have access to a parallel corpus which provides Hindi-English sentence pairs which mean the same. We then use the TextBlob API to convert the English sentence to Hindi and then, label the two Hindi sentences as paraphrases if they are not identical. This labeling is noisy since the translator may make errors. However, with high probability, the sentences are paraphrases. For Malayalam, we convert both the English and the Hindi sentence to Malayalam. This gives us 2 Malayalam sentences which are similarly labeled as positive.

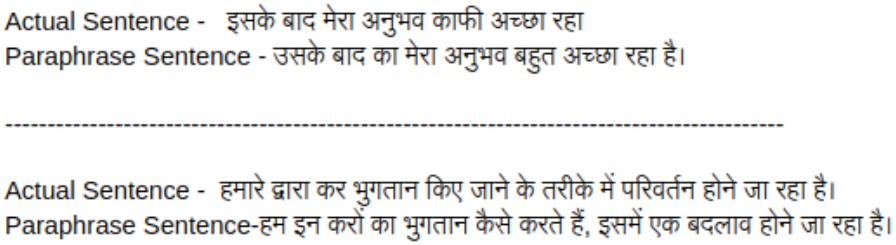


Figure 1: Translation

[1] https://github.com/google-research-datasets/paws
[2] https://github.com/google-research-datasets/paws

## 4.2 Synonym Substitution

We used the Indo-WordNet to get synonyms for a word on the basis of its POS tag. Then, a word in the input sentence is replaced by its synonym to get a paraphrase. To get tagged sentences, we are using StanfordNLP parser. Proper Nounds(NNP) are not replaced. the Pyiwn [3] package is used to get synonyms.

Similar method could not be employed for Malayalam because there was no usable POS tagger for Malayalam. Without POS tag, the synonym substitution using Pyiwn created a large number of errors and was therefore, discarded.

```
Actual Sentence - इस तरह यहाँ की कोई खास शैली ना होकर एक अद्भुत मिश्रण हो गया है।
Paraphrase - इस प्रकार यहाँ की कोई असामान्य ढंग ना होकर एक विचित्र सम्मिश्रण हो गया है ।
--------------------------------------------------
Actual Sentence - यह स्थान भी युनेस्को की संसार अमानत का अंग है।
Paraphrase - यह जगह भी युनेस्को की दुनिया अपमान का अवयव है।
--------------------------------------------------
Actual Sentence - परिणामतः शरीर के समस्त अवयव और अततः सारा शरीर पांचभौतिक है।
Paraphrase - परिणामतः बदन के कुल अंग और अततः सारा बदन पांचभौतिक है।
--------------------------------------------------
```

Figure 2: Synonyms Substitution

## 4.3 Conjunction Order Change

Sentences like - "I like A , B and C" . Here B and C are dependent on A with dependency relation 'CONJ' , so changing order of these words.
**Input Sentence**- "I like A, B and C"
**Output Sentence**- "I like B, C and A"

```
Actual Sentence - भक्तपुर के दरबार स्क्वैयर का निर्माण 16वीं और 17वीं शताब्दी में हुआ था।
Paraphrase - भक्तपुर के दरबार स्क्वैयर का निर्माण 17वीं और 16वीं शताब्दी में हुआ था ।
--------------------------------------------------
Actual Sentence - इसके अनेक संस्करण एवं भाष्य भी प्रकाशित हो चुके हैं ।
Paraphrase - इसके अनेक भाष्य एवं संस्करण भी प्रकाशित हो चुके हैं ।
--------------------------------------------------
Actual Sentence - मुझे सेब , अनार और आम बहुत पसंद हैं।
Paraphrase - मुझे अनार , आम और सेब बहुत पसंद हैं ।
--------------------------------------------------
```

Figure 3: Conjunction Order Change

## 4.4 Proper Noun (NNP) Swapping (Negative)

To generate negative pairs, we are doing proper Noun Swapping.
**Input Sentence**- "A to B"
**Output Sentence**- "B to A"

Cannot be done for Malayalam since we don't have a usable POS tagger.

[3] https://github.com/riteshpanjwani/pyiwn

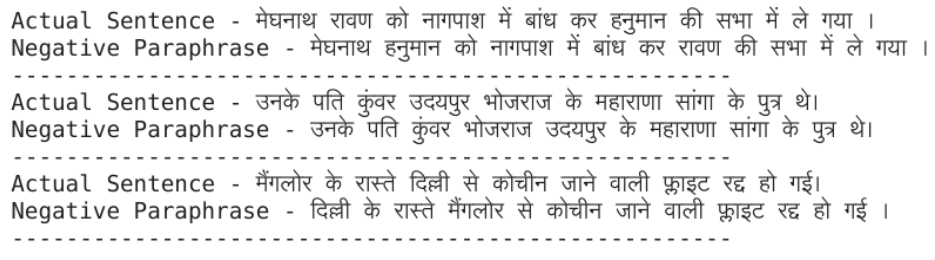


Figure 4: Proper Noun (NNP) Swapping

## 4.5 Explicitly Negation of Sentences

**Observed Rule :** If a word with **POS tag 'NOUN'** has **dependency relationship 'compound'** with its parent and **next word** after it has **POS tag 'VERB'** then we can put **'No/Nahin'** after current word to generate sentence negation.

Similar approach cannot be done for Malayalam since we don't have a good dependency parser for Malayalam. Instead a heuristic is used where if the sentence ends with "Aanu" (is), then it is replaced with "Alla"(is not) which, on inspection, gave correct negatives.

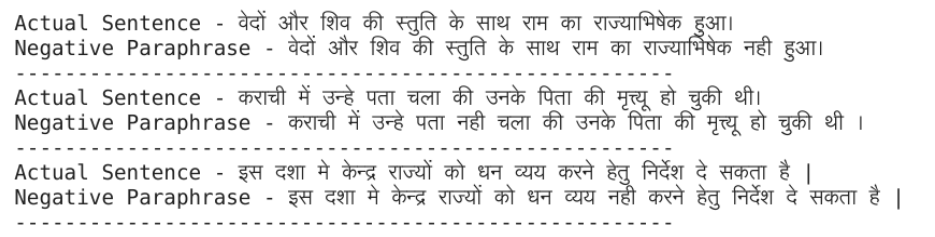


Figure 5: Explicitly Negation of Sentences

## 4.6 Automated Back Translation (for Malayalam)

**Back translation** is a method used to evaluate the quality of translations. In this method, a sentence is translated from a source language S into a target language T and then back to S. Then, the two sentences in the source language can be compared.

To generate paraphrases, we follow these steps:

1. Take a Malayalam sentence $s$
2. Convert s to English using TextBlob API and get $t$
3. Convert $t$ back to Malayalam and get $\hat{s}$
4. If $s \neq \hat{s}$ are not identical, add pair (s, $\hat{s}$) as a positive paraphrase sample.

## 4.7 Agglutination-based paraphrasing using "Sandhi" rules

Malayalam is highly agglutinative and many words can be concatenated together and still give valid sentences. These transformations follow rules called "Sandhi" rules and there are 4 of them native to Malayalam and a few inherited from Sanskrit. Paraphrases can be created by using these rules to join words together. We have implemented one such case and other rules are deferred to future work.

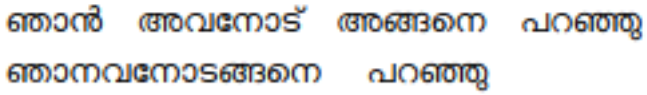


Figure 6: Malayalam sentences which mean "I told him so." 3 words in first sentence have been combined into one word in second version.

### 4.8 Word replacement with pre-specified non-synonyms (for Malayalam negative pairs)

We manually specify some sets of non-synonyms which can be used in same context. Then, a word in set can be replaced by another word in this set to get a negative paraphrase sample.

## 5 Results

- Hindi
  - Positive samples
    - Translation: ∼3044 pairs
    - Synonym substitution: ∼967 pairs
    - Conjunction order change: ∼956 pairs
  - Negative Samples
    - Proper noun swapping:∼990 pairs
    - Explicit negation of sentences:∼974 pairs
- Malayalam
  - Positive samples
    - Translation - ∼566 pairs

## 6 Error Analysis

- **Proper Noun Swapping :** Fails when Proper Nouns have different types like in below sentence, one is Place and other is person.

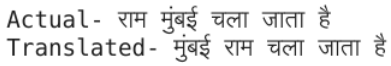


Figure 7: Proper Noun Swapping

- **Conjunction order change:** In some sentences ,changing order of conjunctions changes meaning, resulting in negative pair.

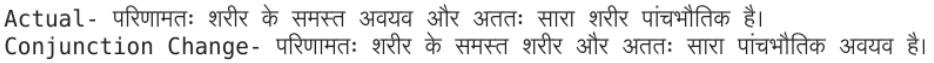


Figure 8: Conjunction order change