Identification, Classification and Alignment of Clauses

Project submitted to

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

In partial fulfillment of the requirements for the degree of

MASTER OF COMPUTER APPLICATIONS, 2019

BY

Sandip Sanpui

Examination Roll: MCA196023

Registration No: 137334 of 2016-2017

Under the guidance of

Prof. (Dr.) Dipankar Das

Assistant Professor, Department of Computer Science &Engineering

Jadavpur University

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

TO WHOM IT MAY CONCERN

I hereby recommend that the project entitled "Identification, Classification and Alignment of Clauses" prepared under my supervision and guidance at Jadavpur University, Kolkata by SANDIP SANPUI *(Reg. No. 137334 of 2016 – 17, Examination Roll No. MCA196023), may be accepted in partial fulfillment for the degree of Master of Computer Applications in the Faculty of Engineering and Technology, Jadavpur University, during the academic year 2018 – 2019. I wish him every success in life.*

……

Prof. (Dr.) Dipankar Das
Project Supervisor,
Department of Computer Science and Engineering
Jadavpur University, Kolkata – 700032.

…….

Prof. (Dr.) Mahantapas Kundu
Head of the Department
Department of Computer Science and Engineering
Jadavpur University, Kolkata – 700032.

……

Prof. (Dr.) Chiranjib Bhattacharjee
Dean, Faculty Council of Engg. & Tech.
Jadavpur University, Kolkata – 700032.

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC PROJECT

I hereby declare that this project contains literature survey and original research work by the undersigned candidate, as part of her MASTER OF COMPUTER APPLICATIONS studies. All information in this document have been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material results that are not original to this work.

NAME: SANDIP SANPUI

ROLL NUMBER: 001610503027

PROJECT TITLE: Identification, Classification and Alignment of Clauses

SIGNATURE WITH DATE

JADAVPUR UNIVERSITY

FACULTY OF ENGINEERING AND TECHNOLOGY

CERTIFICATE OF APPROVAL

The forgoing project is hereby accepted as a credible study of an engineering subject carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve the project only for the purpose for which it is submitted.

FINAL EXAMINATION FOR
EVALUATION OF PROJECT: 1._____

2._____

(Signature of Examiners)

ACKNOWLEDGEMENT

I express my honest and sincere thanks and humble gratitude to my respected teacher and guide *Prof. (Dr.) Dipankar Das*, Assistant Professor, Department of Computer Science & Engineering, Jadavpur University, for his exclusive guidance and entire support in completing and producing this project successfully. I am very much indebted to him for the constant encouragement, and continuous inspiration that he has given to me. The above words are only a token of my deep respect towards him for all he has done to take my project to the present shape.

I would like to thank *Mr. Sainik Kumar Mahata* for valuable support and suggestions to the activities of the project.

Finally, I convey my real sense of gratitude and thankfulness to my family members, for being an endless source of optimism and positive thoughts; and last but not the least, my father & mother for their unconditional support, without which I would hardly be capable of producing this huge work.

Sandip Sanpui
Examination Roll: MCA196023
Registration No: 137334 of 2016 – 2017

Content

Abstract

Parallel corpus is a collection of bilingual sentence pair where every sentence is a translation of the other. Such corpus is very essential for a **Machine Translation** system to produce quality output. Moreover, it is well documented that using **Simple Sentences** only, while training a Machine Translation system, produces better output. But, while parallel corpus for various language pairs are abundant, parallel corpus consisting only simple sentences are rare. The purpose of the current work is to build a English-Bengali parallel corpus comprising of simple sentences only. We tend to make use of Bengali, as it is a low-resource language and parallel corpus of large size, albeit consisting only simple sentences, are not readily available. We have used Google Translate API for the translation task. Also, we have devised a method by which we can classify English sentences as simple or **Other (Complex/Compound)**. A provision is also made to simplify the complex/compound sentences to the simple sentence has been proposed. This method is based on **Clause Boundary Identification.** Finally, a basic alignment technique has been implemented to align the segments of English with respect to their corresponding segments in Bengali by utilizing basic punctuation markers and conjuncts and/or disjuncts. The proposed techniques will be useful for identifying, classifying and aligning sentences in a parallel corpus containing English-Bengali translation pairs.

Keywords: Clause Boundary Identification; Machine Translation; Parallel Corpus; Simple Sentence; Complex/Compound Sentence; Segmentation; Alignment

1. Introduction

1.1. Problem Statement

Machine **T**ranslation (MT) is an automated process, which translates from one natural language, called the source language, to another natural language, called the target language. Users can use this service for translating one language to another. MT is from the broad area of **A**rtificial **I**ntelligence (AI) and **N**atural **L**anguage **P**rocessing (NLP) and training of such systems, rely heavily on large and good quality bilingual parallel corpus. Now, parallel corpus are abundant for languages spoken by majority of the human population. But, low-resourced languages, that are not spoken widely, have lower digital footprint. This leads to low parallel sentence count, considering parallel corpora. **So, creating parallel corpus for low-resourced languages have always been a research problem.**

Moreover, we have wide research evidence, that when a MT system is trained using simple sentences only, it leads to better translation output. This is due to the fact that simple sentences are semantically lucid and a system, when trained using these, can easily extract the nuances of the language. **Since, parallel corpus of low-resourced languages are hard to find, it becomes even more difficult, to construct a parallel corpus of low-resourced language, comprising of simple sentences only.**

Initially, to identify simple, complex and compound sentences, Clause Identification is necessary. Clause identification is a special kind of dependency parsing, like text chunking Nevertheless, it is more difficult than text chunking, since clause can have embedded clauses. Clauses information is important for several more elaborated tasks such as full parsing and semantic role labeling. **While, clause identification can be done using state-of-art methods for English, but it is quite difficult to do for low-resourced languages as standard features, lexicons and tools are not readily available. This becomes another research challenge that we have to cater to.**

Moreover, after detecting complex/compound sentences, we have to simplify these to two or more simple sentences using Clause Boundary Identification. **Clause boundary identification of natural language sentences poses considerable difficulties due to the ambiguous nature of natural languages. Again, this may be an easy task for English language, but is quite trivial for low-resourced languages as standard libraries aren't available for the same.**

1.2. Literature Survey

Various works has already been done on Dependency Parsing, clause boundary identification, text simplification, machine translation and generation of parallel corpus.

Christopher D Manning et. al. [1] worked on describe the design and use of the Stanford CoreNLP toolkit, an extensible pipeline that provides core natural language analysis. This toolkit is quite widely used, both in the research NLP community and also among commercial and government users of open source NLP technology. We suggest that this follows from a simple, approachable design, straightforward interfaces, the inclusion of robust and good quality analysis components, and not requiring use of a large amount of associated baggage.

Marie-Catherine De Marneffe et.al. [2] worked on the paper examines the Stanford typed dependencies representation, which was designed to provide a straightforward description of grammatical relations for any user who could benefit from automatic text understanding. For such purposes, we argue that dependency schemes must follow a simple design and provide semantically contentful information, as well as offer an automatic procedure to extract the relations. We consider the underlying design principles of the Stanford scheme from this perspective, and compare it to the GR and PARC representations. Finally, we address the question of the suitability of the Stanford scheme for parser evaluation.

Matthew Shardlow [3] worked on text simplification modifies syntax and lexicon to improve the understandability of language for an end user. This survey identifies and classifies simplification research within the period 1998-2013. Simplification can be used for many applications, including: Second language learners, preprocessing in pipelines and assistive technology. There are many approaches to the simplification task, including: lexical, syntactic, statistical machine translation and hybrid techniques. This survey also explores the current challenges, which this field faces. Text simplification is a non-trivial task, which is rapidly growing into its own field. This survey gives an overview of contemporary research whilst taking into account the history that has brought text simplification to its current state.

Avinesh.PVS et. al. [4] worked on Chunking using Conditional Random Fields (CRFs) and Transformation Based Learning (TBL) for Telugu, Hindi and Bengali. They showed that training CRFs can help to achieve good performance over any other Machine Learning (ML) techniques. Improved training methods based on the morphological information, contextual and the lexical rules (developed using TBL) were critical in achieving good results. The CRF and TBL based POS tagger has an accuracy of about 77.37%, 78.66%, and 76.08% for Telugu, Hindi and Bengali, and the chunker performs at 79.15%, 80.97% and 82.74% for Telugu, Hindi and Bengali respectively.

EliorSulem et. al. [5] worked on Sentence splitting is a major simplification operator. Here we present a simple and efficient splitting algorithm based on an automatic semantic parser. After splitting, the text is amenable for further fine-tuned simplification operations. In particular, we show that neural Machine Translation can be effectively used in this situation. Previous application of Machine Translation for simplification suffers from a considerable disadvantage in

that they are over-conservative, often failing to modify the source in any way. Splitting based on semantic parsing, as proposed here, alleviates this issue. Extensive automatic and human evaluation shows that the proposed method compares favorably to the state-of-the-art in combined lexical and structural simplification.

Erik F. Tjong Kim Sang et. al. [6] worked on dividing texts into syntactically related non-overlapping groups of words, a so-called text chunking. They gave background information on the data sets, presented a general overview of the systems and discussed their performance.

Sarah E. Petersen et. al. [7] worked on text simplification for language learners. Teachers and students in bilingual education and other language-learning contexts commonly use Simplified texts. Their goal was the development of tools to aid teachers by automatically proposing ways to simplify texts. Their paper presents a detailed analysis of a corpus of news articles and abridged versions written by a literacy organization in order to learn what kinds of changes people make when simplifying texts for language learners.

Claire Cardie et. al. [8] found out that finding simple, non-recursive, base noun phrases are an important subtask in many natural language processing applications. They presented a corpus-based approach for finding base NPs by matching part-of- speech tag sequences. The training phase of the algorithm was based on two successful techniques: first the base NP grammar is read from a Treebank corpus; then the grammar is improved by selecting rules with high benefit scores. Using this simple algorithm with a naive heuristic for matching rules, they achieved surprising accuracy in an evaluation on the Penn Treebank Wall Street Journal.

R. Vijay Sundar Ram et. al.[9] worked on the detection of clause boundaries using a hybrid approach. The Conditional Random fields (CRFs), which have linguistic rules as features, identified the boundaries initially. The boundaries marked were checked for false boundary marking using Error Pattern Analyzer. The false boundary markings were re-analyzed using linguistic rules. The experiments done with their approach showed encouraging results and is comparable with the other approaches.

Erik F. Tjong Kim Sang et. al.[10] used seven machine learning algorithms for one task: identifying base noun phrases. The results were processed by different system combination methods and all of these outperformed the best individual result. They have applied the seven learners with the best combinatory, which is a majority vote of the top five systems to a standard data set and managed to improve the best published result for this data set.

Kerstin Denecke[11] introduced a methodology for determining polarity of text within a multilingual framework. The method leveraged on lexical resources for sentiment analysis available in English SentiWordNet. First, a document in a different language than English was translated into English using standard translation software. Then, the translated document was classified according to its sentiment into one of the classes “positive” and “negative”. For sentiment classification, a document is searched for sentiment bearing words like adjectives. By means of SentiWordNet, scores for positivity and negativity were determined for these words. An interpretation of the scores then led to the document polarity. The method was tested for German movie reviews selected from Amazon and is compared to a statistical polarity classifier

based on n-grams. The results showed that working with standard technology and existing sentiment analysis approaches was a viable approach to sentiment analysis within a multilingual framework.

Federico Zanettin [12] worked on how small bilingual corpora of either general or specialized language can be used to devise a variety of structured and self-centered classroom activities whose aim was to enhance the understanding of the source language text and the ability to produce fluent target language texts.

Colin Bannard et. al. [13] worked on Using alignment techniques from phrase based statistical machine translation, they showed how paraphrases in one language can be identified using a phrase in another language as a pivot. They define a paraphrase probability that allows paraphrases extracted from a bilingual parallel corpus to be ranked using translation probabilities, and show how it can be refined to take contextual information into account. They have evaluated their paraphrase extraction and ranking methods using a set of manual word alignments, and contrast the quality with paraphrases extracted from automatic alignments.

Daniel Varga et. al.[14] worked on e a general methodology for rapidly collecting, building, and aligning parallel corpora for medium density languages, illustrating their main points on the case of Hungarian, Romanian, and Slovenian. They have also described and evaluated the hybrid sentence alignment method, which they are using.

Sabine Buchholz et. al. [15] worked on We describe the CoNLL-2000 shared task: dividing text into syntactically related non-overlapping groups of words, so-called text chunking. We give background information on the data sets, present a general overview of the systems that have taken part in the shared task and briefly discuss their performance.

Constantin Orasan [16] proposed a hybrid method for clause splitting in unrestricted English texts, which required less human work than existing approaches. A shallow rule-based module processed the results of a machine-learning algorithm, trained on an annotated corpus, in order to improve the accuracy of the method. The evaluation of the results showed that the machine-learning algorithm is useful for identification of clause's boundaries and the rule-based module improved the results. Using some very simple rules they reported precision of around 88%.

JoakimNivre et. al. [17] The Conference on Computational Natural Language Learning features a shared task, in which participants train and test their learning systems on the same data sets. In 2007, as in 2006, the shared task has been devoted to dependency parsing, this year with both a multilingual track and a domain adaptation track. In this paper, we define the tasks of the different tracks and describe how the data sets were created from existing treebanks for ten languages. In addition, we characterize the different approaches of the participating systems, report the test results, and provide a first analysis of these results.

AdvaithSiddharthan et. al. [18] worked on a framework for text simplification based on applying transformation rules to a typed dependency representation produced by the Stanford parser. We test two approaches to regeneration from typed dependencies:(a) gen-light, where the

transformed dependency graphs are linearised using the word order and morphology of the original sentence, with any changes coded into the transformation rules, and (b) gen-heavy, where the Stanford dependencies are reduced to a DSyntS representation and sentences are generating formally using the RealPro surface realiser. The main contribution of this paper is to compare the robustness of these approaches in the presence of parsing errors, using both a single parse and an n-best parse setting in an over generate and rank approach. We find that the gen-light approach is robust to parser error, particularly in the n-best parse setting. On the other hand, parsing.

2. Methodology

2.1. Generation of English sentences corpora

A wide array of different types of parallel corpora has been constructed for use in the field of MT. They reflect the criteria according to which they are designed and the purpose for which they are developed. Such a corpus used in translation is a bilingual corpus. Language pairs are put together on the basis of "parallelism". Parallel bilingual corpora consist of texts in language *"A"* and their translation into language *"B"*, or vice versa. The relationship between texts is directional, i.e. it goes from one text; the source language (SL) text to another text; the target language (TL) text. To prepare such a parallel corpus for English-Bengali language pair, we collected 49,999 English-Bengali parallel coprus from **T**echnology **D**evelopment for **I**ndian **L**anguages Programme[1] (TDIL). In addition to this, we collected 57,985 English sentences from the resource of **M**achine **T**ranslation in **I**ndian **L**anguages (MTIL) shared task[2], organized by Amrita University. Similarly, 7,053 English sentences from various other websites and the statistics are shown in Table 1.

Source	Data Size
Various Websites	7,053
TDIL	49,999
Amrita University	57,985
Total	1,15,037

Table 1: Data Information Table

2.2. Translation using Google Translate API

We translated 65,038 English sentences into Bengali using Google Translator API[3] for Python. Then, the English sentences and their corresponding Bengali translations are aligned in parallel, to produce an English-Bengali parallel corpus of 1,15,037 sentences.

2.3. Pre-Processing

2.3.1. Shallow Parsing/Chunking

Syntactic Parsing or **Chunking** is the task of recognizing a sentence and assigning a syntactic structure to it. The most widely used syntactic structure is the parse tree which can be generated

[1]http://tdil.meity.gov.in/

[2]http://nlp.amrita.edu/mtil_cen/

[3]https://pypi.org/project/googletrans/

using some parsing algorithms. These parse trees are useful in various applications like grammar checking or more importantly it plays a critical role in the semantic analysis stage. For example to answer the question "*Who is the point guard for the LA Laker in the next game ?*" we need to figure out its subject, objects, attributes to help us figure out that the user wants the point guard of the LA Lakers specifically for the next game.

Shallow Parsing is an analysis of a sentence in which constituent parts of sentences (nouns, verbs, adjectives, etc.) are identified and then higher order units that have discrete grammatical meanings (noun groups or phrases, verb groups, etc.) are linked. While the most elementary parsing algorithms simply link constituent parts on the basis of elementary search patterns (e.g. as specified by Regular Expressions), approaches that use machine learning techniques (classifiers, topic modeling, etc.) can take contextual information into account and thus compose parses in such a way that they better reflect the semantic relations between the basic constituents. We have used Natural Language Toolkit (NLTK)[4] and Stanford Parser[5] for performing the shallow parsing on the English sentences. We avoided shallow parsing the Bengali sentences as no standard library were available for the same. Example of shallow parsing is given in Table 2.

Sentence before parsing	**After dependency parsing**
By drinking plenty of water not only the left-over pieces of food gets cleaned but saliva also gets formed.	S (PP (IN By) (S (VP (VBG drinking) (NP (NP (NP (RB plenty)) (PP (IN of) (NP (NN water)))) (CONJP (RB not) (JJ only)) (NP (DT the) (NN left-over)))))) (NP (NP (NNS pieces)) (PP (IN of) (NP (NN food)))) (VP (VBZ gets) (SBAR (S (NP (NNP cleaned) (CC but) (NNP saliva))(ADVP (RB also)) (VP (VBZ gets) (ADJP (VBN formed)))))) (. .))
Taking a spoon of salt pour three to four drops of lemon juice in that.	S (S (VP (VBG Taking) (NP (NP (DT a) (NN spoon)) (PP (IN of) (NP (NN salt)))))) (VP (VBP pour) (NP (NP (QP (CD three) (TO to) (CD four)) (NNS drops)) (PP (IN of)(NP (NP (JJ lemon) (NN juice)) (PP (IN in) (NP (DT that))))))) (. .))

Table 2: Example of tagging by Shallow parsing

[4] https://www.nltk.org/

[5] http://nlp.stanford.edu:8080/parser/

2.4. Sentence Identification

2.4.1. Identification of Simple and Complex/Compound Sentences

A simple sentence in this context is defined as a sentence, which contains only one independent clause and has no dependent clauses. Generally, whenever two or more clauses are joined by conjunctions (coordinating conjunction and subordinating conjunction), it becomes a complex or a compound sentence accordingly.

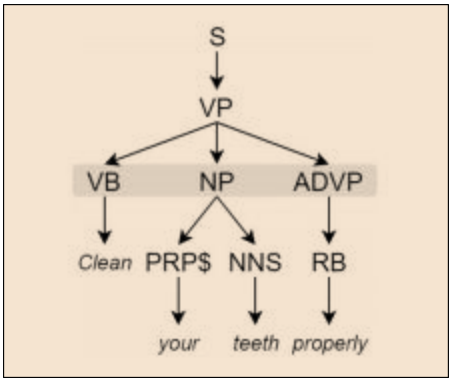


Figure 1: Extraction of phrase chunks.

We noticed that, simple, complex and compound sentences have a unique phrase structure that consists of combinations of NP, VP, ADVP and PP. In conjunction to this theory, we applied a machine learning based approach to extract sentences of various complexities from the English corpus.

We subjected 3,046 simple sentences, 2,698 complex sentences and 3,547 compound sentences to shallow parsing, and extracted the unique phrase structures. These sentences were collected from various web sources. This constituted the rules by which we further mined for sentences of various complexities from the English corpus. We extracted 205 unique rules for simple sentences, 176 unique rules for complex sentences and 215 unique rules for compound sentences. The surface forms of the rules along with their confidence score, are shown in Table 3, 4 and 5 respectively. Confidence Score was calculated as a fraction of total number of sentences identified using a specific rule, by the total number of sentences.

$$Confidence = \frac{total\ \#\ sentences\ identified\ using\ specific\ rule}{total\ \#\ of\ sentences}$$

Rules	Confidence Scores
PP NP* PP VP NP*	8.4
PP NP* VP PP NP*	9
ADVP NP* VP* ADVP NP*	9
NP VP PP NP PP NP	**12**
NP ADVP VP* NP*	**12**
NP* VP NP*	**11.69**
NP* PP NP VP* NP	**11.46**
NP VP PP NP*	**11.23**
VP* NP* PRP* ADVP*	4.92
NP VP* NP* PP* ADJP* ADVP*	9.62

Table 3: Rules surface forms for Simple Sentences.

Rules	Confidence Scores
NP* NNP PP* IN* DT* NN* NNS	8.6
S* ADVP RB* SBAR IN NP* PRP* VP* VB* MD DT NN	8.7
S* NP* DT* JJ NN* ADVP RB VP* VBZ VB* PRP* MD	**10.1**
S* NP* NNP NNS VP* VBP* VBN DT* JJ NN* SBAR IN PRP RP* TO PP* ADVP RB IN	8.13
S* NP* NNS* VP* MD RB VB SBAR WHADVP WRB EX VBZ JJ	9.21
S* NP* DT* NN* SBAR WHNP WP$ VP* VBD VBZ	9.16
S* NP* NNP* SBAR WHNP WP VP* VBZ DT JJ VBZ PP IN	7.2
S* NP* PDT DT* NNS VP* VBD* SBAR WHADVP WRB NN*	7.31
S* NP* PRP ADVP RB* VP* VBD* DT* NN* SBAR IN ADJP JJ	**10.29**
S* PP IN VP* VBG ADJP RB JJ NP* PRP VBD TO VB DT NN	8.12
S* NP* NNP VP* VBD* SBAR WHADVP WRB PRP$ NN ADJP JJ CC PRP ADVP RB ADVP RBR	**10.75**
S* VP* VB* NP* PRP* MD ADVP RB NN .	9.62
S* SBAR IN NP* DT NN VP* VBD* ADVP RB PRP ADJP JJ	7.58
S* SBAR IN NP* DT NN VP* VBD* ADVP RB PRP ADJP JJ	7.78
S* VP* VB SBAR WHNP WP VBZ NP* DT NN PP IN JJ NN*	9.65

Table 4: Rules surface forms for *Complex* Sentences.

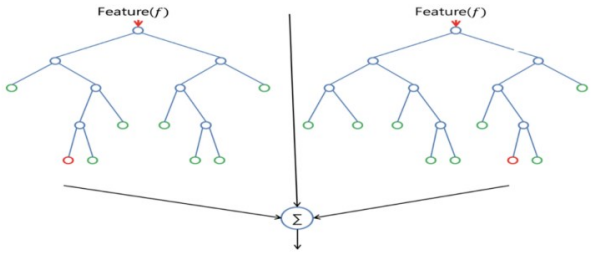
Rules	Confidence Scores
S* CC NP* PRP* VP* VBZ* ADJP JJ CC NN	8.65
S NP DT JJ NN VP* MD RB VB CC VBP	6.26
S NP* NNP* VP* VBD DT NN CC VBN PRP	**11.12**
S* NP* NNS* VP* VBP* JJ* SBAR IN PRP ADJP CC RB ADJP	**11.52**
S NP PRP VP* MD DT VB* ADVP RB CC RP	**10.86**
S NP PRP VP* MD DT VB* ADVP RB CC RP	**12.87**
S* NP* DT NN* VP* VBD* PP IN DT* JJ CC NNS PRP ADVP RB	**11.54**
S* NP* DT* NN* VP* VBD* PP IN JJ CC NNS PRP ADVP RB	**11.53**
S* NP* NN* VP* VBD* ADJP JJ IN PRP PP TO DT ADVP RB	8.69
S* NNP VP MD VP* VB VBN PP IN NP* NNS CC PRP VBZ DT JJ NN	8.47
S* NP* NNS* VP* VBP* JJ TO VB* PRP* MD NN* SBAR PP IN CD	**12.68**
S* NP* PRP* ADVP RB* VP* VBP* TO* VB* TO CC ADJP JJ	7.78
S* NP* PRP$ NN* VP* VBD* ADJP JJ CC PRP DT	**11.92**
S* NP* VB* VP* VBP* NN* PRP MD SBAR IN DT VBN	**10.25**
S* VP* VBP ADVP RB* NP* PRP MD PRP	8.91

Table 5: Rules surface forms for *Compound* Sentences.

The rules, along with their respective labels were trained using Decision Tree, Random Forest and Naive Bayes classifier.

2.4.2. Random Forest Classifier

Random forests or **random decision forests** are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision tree at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set.



Random Forest Algorithm

We tested our system on 2876 sentences (1438 simple sentences and 1438 complex/compound sentences) and achieved an accuracy of 78.22%.

2.4.3. Naive Bayes Classifier

Naive Bayes classifier is a probabilistic machine learning model that's used for classification task. The crux of the classifier is based on the Bayes theorem.

Bayes Theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Using Bayes theorem, we can find the probability of **A** happening, given that **B** has occurred. Here, **B** is the evidence and **A** is the hypothesis. The assumption made here is that the predictors/features are independent. That is presence of one particular feature does not affect the other. Hence it is called naïve. We tested our system on 2876 sentences (1438 simple sentences and 1438 complex/compound sentences) and obtained an accuracy of 79.96%.

2.4.4. Decision Tree Classifier

Decision Trees are a type of Supervised Machine Learning (that is you explain what the input is and what the corresponding output is in the training data) where the data is continuously split according to a certain parameter. The tree can be explained by two entities, namely decision nodes and leaves. The leaves are the decisions or the final outcomes. And the decision nodes are where the data is split. We tested our system on 2876 sentences (1438 simple sentences, 719 complex sentences and 719 compound sentences) and achieved an accuracy of 84.84%. The confusion matrix for the same is given below.

	Simple	**Complex**	**Compound**	**Precision**
Simple	1275	72	81	89.83%
Complex	75	572	45	82.66%
Compound	88	75	593	78.44%
Recall	88.67%	79.56%	82.48%	
Accuracy	84.84%			
Kappa	0.758			

Table 6: Confusion Matrix and accuracy figures for system developed using Decision Tree algorithm.

2.5. Alignment Framework

Simplification consists of modifying the content and structure of a text in order to make it easier to read and understand, while preserving its main idea and approximating its original meaning. A simplified version of a text could benefit low literacy readers, English learners, children. Also, simplifying a text automatically could improve performance on other NLP tasks, such as parsing, summarization, information extraction, and machine translation.

Sentence simplification is the process of simplifying the complex sentences into simpler sentences. The method proposed in this paper is simpler one. Based on rules, the sentences are simplified in order to get exact translation. When a clause stands on its own and is independent, it is called main clause. Subordinate clauses are those clauses which cannot stand alone but depend onmain clause for their meaning. Most of the sentences contain conjunctions and sentences are split based on conjunctions. Independent clauses can be joined by a coordinating conjunction to form complex or compound sentences. Dependent clauses often begin with a subordinating conjunction or relative pronoun.

Our system handles coordinating conjunctions, subordinating conjunctions and relative pronouns. Coordinating conjunction includes for, and, not, but, or, yet and so. Subordinating conjunction includes after, although, because, before, if, since, that, though, unless, where, wherever, when, whenever, whereas, while, why. Relative pronoun includes who, which, whose, whom.

2.5.1 Segmentation

The proposed approach follows in following steps:

- Split the sentences from the paragraph based on delimiters such as "." and "?"
- Delimiters such as (comma, {,}, [,],) are ignored from the sentences.
- Individual sentences are split based on coordinating and subordinating conjunction.

The text can be of any form i.e., paragraphing format, individual sentences, etc. Presence of delimiter such as (? and .), is an important pre-requisite as the initial splitting is done based on delimiters. The obtained individual sentences are parsed using Stanford parser. Stanford Parser gives POS tag as well as dependency information; based on the information the rules are generated.

Our system deal with the following techniques,

- Splitting
- Simplification

There are several "wh" connectives available out of which "who, whom, which, whose" are dealt. In this case, the relative clause can occur either in between the main clause, or after the main clause. In both the cases, the connective words contain two possible dependency tags i.e. either "subject" or "object".

Our work is sentence simplification. The simplifying sentences will work for any translation system with English as source language and transfer to Bengali as target language.

Here, “who” is the subordinating conjunction. The sentence should be simplified based on “who”. In the above example, two words are present before “who”. Make ensure that any of these words contain “verb” tag. If so, then the sentence is not embedded within the main clause. But in the above sentence, the parsed information of the first two word “The” and “people” is {DT, NNS}. So this indicates that the relative clause is embedded within the main clause.

Input Sentence:

- It was Partha **who** paid for the drinks.

Output Sentence:

- It was Partha.
- who paid for the drinks .

In this case, the sentences are split based on the conjunctions. Coordinating conjunction includes (for, and, not, but, or, yet, so) and POS tag for coordinating conjunction is **“CC”** and the dependency tag is “cc”. Subordinating conjunction includes (when, whenever, where, wherever, if, because, unless, though,etc.). Here, the relative clause can occur before the main clause, or after the main clause.

Consider an example,

Input Sentence:

- She returned the computer after she noticed it was damaged.

After Parsing:

S (NP (PRP She)) (VP (VBD returned) (NP (DT the) (NN computer)) (SBAR (IN after) (S (NP (PRP she)) (VP (VBD noticed) (SBAR (S (NP (PRP it)) (VP (VBD was) (VP (VBN damaged))))))))) (. .))

In the above example relative clause is present after the main clause. Here, *„but"* is the coordinating conjunction and it is the splitter word. Here the sentences will be split into two simple sentences based on the splitter word. The connective word is always present in the relative clause.

Output Sentence:

- She returned the computer.
- After she noticed it was damaged.

Example for conjunction .

1.We are going to the school but the school was closed.

We are going to the school

but the school was closed .

2.Ram cried when his dog got sick but he soon got better.

Ram cried when his dog got sick

but he soon got better .

3. They got there early, and they got really good seats.

They got there early,

And they got really good seats.

4.He gave up trying because he did not succeed.

He gave up trying.

Because he did not succeed.

5.Although he was wealthy still he was unhappy.

Although he was wealthy.

Still he was unhappy.

6.She must weep, or she will die.

She must weep.

Or she will die.

2.5.2 Alignment

We can clearly see that complex and compound sentences can be split into two or more simple sentences, using

1. Delimiters (DL)
2. Coordinating Conjunctions (CC)
3. Subordinating Conjunctions (SC)

We found out that, English follows the S-V-O syntactic structure and Bengali follows the S-O-V syntactic structure. This means that positions of Nouns and Verbs can change when comparing English and Bengali, but the positions of the DL, CC and SC do not change. Our hypothesis was that if we can translate the CC and SC (DL to some extent), we can split the Bengali sentences as well. Examples of such segmentation are given below.

Input Sentence:

Rabi waited for the train, but the train was late .

In Bengali font:

রবি ট্রেনের জন্য অপেক্ষা করছিলেন, কিন্তু ট্রেন দেরি হয়ে গেল।

Output in English:

Rabi waited for the train .

but the train was late .

Output in Bengali:

রবি ট্রেনের জন্য অপেক্ষা করছিলেন।

কিন্তু ট্রেন দেরি হয়ে গেল।

Input Sentence:

The sky is clear; the stars are twinkling.

In Bengali font:

আকাশ পরিষ্কার; তারা জ্বলছে।

Output in English: The sky is clear.

The stars are twinkling.

Output in Bengali:

আকাশ পরিষ্কার।

তারা জ্বলছে।

3. Result and Evaluation

3.1. Result

We have collected 1,15,037 (49,999 parallel English-Bengali data) English sentences from TDIL, Amrita University and various web sources. The English sentences were translated and a parallel corpus was developed. For identifying the Simple, Complex and Compound sentences from our parallel corpus, the sentences were parsed and unique rules for each comlexity was found out. We used various machine learning algorithms to train a system that automatically perform the classification. The system trained using Decison Tree classification algorithm gave us the maximum accuracy of 84.44%.

The collected 1,15,037 English sentences were classified using the same model and the classification results are shown in Table 7. It is to be noted that, since we could not parse the corresponding Bengali sentences, we considered the corresponding Bengali sentences as having the same complexity as that of the source English sentence.

Type of Sentence	**# of sentence**
Simple	16,654
Complex	39,068
Compound	50,756
Untagged	8,559

Table 7: Classification System result.

After text simplification and allignment was done, the complex and compound sentences were split into two or more simple sentences, using methods described in Section 2.5. Hence, there was an increment in the size of the parallel corpus. The size of corpus before and after text simplification is shown in Table 8.

Condition	**Type**	**No. Of Sentences**
Before Split	**Complex**	39,068
	Compound	50,756
After Split	**Complex**	71,256
	Compound	89,947
Errors	**Complex**	3,867
	Compound	6,726

Table 8: Result after Sentence splitting.

4. Conclusion

Machine translation systems generate high-quality translation, when trained using Simple sentences, compared to, when trained using sentences of varying complexities. But, parallel corpus for low resourced languages are rare. On top of that, parallel corpus for low resourced language, comprising only of simple sentences are very hard to find. This research challenge led us to developing the same. The classification system can still be improved as using some additional features would lead to better machine learning. Similarly, the rules for splitting complex/compound sentences into two or more simple sentences can be enriched as well. Most importantly, it is to be noted that throughout the reported work, we have used parsing for English sentences only. This is due to the absence of a standard, well-received parser for Bengali. This leads us to believe that, the whole process will become easier, if such a parser can be developed.

Bibliography

[1] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J Bethard, David McClosky, “The Stanford CoreNLP Natural Language Processing Toolkit” 2014

[2] Marie-Catherine De Marneffe, Christopher D Manning Marie-Catherine De Marneffe, Christopher D Manning, “The Stanford typed dependencies representation” 23-08-2008

[3] Matthew Shardlow, “A Survey of Automated Text Simplification” 2014

[4] PVS Avinesh., Karthik G.Part-Of-Speech Tagging and Chunking using Conditional Random Fields and Transformation Based Learning, In the Proceedings of Shallow Parsing for South Asian Languages, pages 21-24,2007

[5] EliorSulem, Omri Abend, Ari Rappoport, “Simple and effective text simplification using semantic and neural methods” 11-10-2018

[6] Sang Erik F. TjongKim , Sabine Buchholz.Introduction to the CoNLL-2000 Shared Task: Chunking, In the Proceedings of CoNLL-2000 and LLL-2000, pages 127–132, Lisbon, Portugal, pages 127-132, 2000

[7] Petersen Sarah E., Mari Ostendorf.Text Simplification for Language Learners: A Corpus Analysis, Speech and Language Technology in Education (SLaTE 2007),pages 69-72, 2007

[8] Cardie Claire, David Pierce.Error-Driven Pruning of Treebank Grammars for Base Noun Phrase Identification,In the Proceeding COLING '98 Proceedings of the 17th international conference on Computational linguistics - Volume 1,Proceeding ACL '98 Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1, pages 218-224,1998

[9] Ram R. Vijay Sundar, Sobha Lalitha Devi.Clause Boundary Identification Using Conditional Random Fields,A.Gelbukh (Ed.): CICLing 2008, LNCS 4919, 2008. © Springer-Verlag Berlin Heidelberg 2008

[10] Sang Erik F. Tjong Kim, Walter Daelemans, Herv´eD´ejean, Rob Koeling, Yuval Krymolowski, VasinPunyakanok, Dan Roth.Applying System Combination to Base Noun Phrase Identification, In the Proceedings of COLING 2000, Saarbru¨cken, Germany, 2000

[11] DeneckeKerstin.UsingSentiWordNet for Multilingual Sentiment Analysis,Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference on, pages 507-512, 2008

[12] Federico Zanettin , “Bilingual comparable corpora and the training of translators” 1998

[13] Chris Callison-Burch, Colin Bannard, Josh Schroeder, “Scaling phrase-based statistical machine translation to larger corpora and longer phrases” 25-06-2005

[14] Bannard Colin, Chris Callison-Burch.Paraphrasing with Bilingual Parallel Corpora, In Proceedings of the 43rd Annual Meeting of the ACL, pages 597–604, Ann Arbor, June 2005. 2005 Association for Computational Linguistics

[15] VargaD´aniel, P´eterHal´acsy ,Andr´asKornai, ViktorNagy, L´aszl´oN´emeth, ViktorTr´on.Parallel corpora for medium density languages, Amsterdam Studies In The Theory And History Of Linguistic Science Series 4, 2007

[16] Resnik Philip ,Noah A. Smith.The Web as a Parallel Corpus, Computational Linguistics, pages 349-380, 2003

[17] Ora˘sanConstantin.A hybrid method for clause splitting in unrestricted English texts, In the Proceedings of ACIDCA’2000, 2000

[18] JoakimNivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, Deniz Yuret, “The CoNLL 2007 shared task on dependency parsing” 2007

[19] Agrawal Himanshu.POS tagging and Chunking for Indian Languages, Shallow Parsing for South Asian Languages,2007

[20] AdvaithSiddharthan , “Text Simplification using Typed Dependencies: A Comparison of the Robustness of Different Generation Strategies” , 23-08-2004

[21] Philip Resnik ’Parallel Strands: A Preliminary Investigation into Mining the Web for Bilingual Text’, in In 3rd Conference of the Association for Machine Translation in the Americas, pages 72–82, Springer, 1998