Towards Discourse Parsing and Connective Identification in Hindi

Thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science
in
Computational Linguistics by Research

by

Sahil Bakshi
20161200
sahil.bakshi@research.iiit.ac.in

International Institute of Information Technology, Hyderabad
(Deemed to be University)
Hyderabad - 500 032, INDIA
August 2022
International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled “Towards Discourse Parsing and Connective Identification in Hindi” by Sahil Bakshi, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Adviser: Prof. Dipti Misra Sharma
To my parents
Acknowledgments

I would like to express my sincerest thanks to my advisor Dr. Dipti Misra Sharma for her constant guidance and support throughout my time as a student at IIIT Hyderabad. She was always an extremely approachable, encouraging, and kind hearted person and I couldn’t have asked for a better mentor to guide me with my research.

I would also like to thank all my professors who taught me over the years, especially Dr. Manish Shrivastava, Dr. Nimmi Rangaswamy and Dr. Radhika Mamidi. Their lectures, as well as the projects I did with them, played a major role in building my interest in the field of NLP. Pruthwik and Vandan, thank you for being so patient with all my tiny, annoying questions and giving me much needed advice throughout my time as a research student in LTRC.

Thank you to all my friends who made my years in IIIT the most enjoyable years of my life. It’s a long list but Aditya, Saurabh, Aniket, Abhishek, Animesh, Vashist, Anish, Kunal, Sumukh, Nikhil, Mohsin, and Tanmai, you guys were always there for me. Thank you for all the fun memories which I will cherish forever.

Lastly and most importantly, I would like to thank my family for always supporting me no matter what. Thank you for all your sacrifices, for financially supporting me throughout my life, and for allowing me to pursue my dreams. I will forever be grateful to you guys.
Abstract

Discourse parsing is a sub-field of natural language processing which involves understanding the structure, information flow, and modeling the coherence of a given text. It forms the basis of several natural language processing tasks, including, but not limited to, question-answering, text summarization, and sentiment analysis. One of the fundamental tasks in discourse parsing is discourse unit segmentation and connective identification. Discourse unit segmentation refers to identifying the elementary units of text that combine to form a coherent text. Connectives signal the presence of explicit discourse relations in text. Connective identification is the task of identifying these discourse connectives.

Language has always played a significant role in human interaction and the evolution of society. With the increasing amount of text data being generated every day on social media platforms such as Facebook, Twitter, WhatsApp, Reddit, etc., helping machines understand and analyse this data is going to be the fundamental task which will further enable us to improve the performance of systems for downstream NLP tasks. In this thesis, we explore the sub-field of shallow discourse parsing, compare approaches towards segmentation and connective identification, and build a dataset and connective identification system for Hindi data.

First, we look at approaches towards shallow discourse parsing to identify individual discourse relations that are present in text. This involves given a text, identifying the span of the explicit discourse connective, labelling the two text spans that act as the arguments of the connective and predicting the sense of the discourse relation. We compare and analyse several approaches for these tasks. We then look at an approach towards shallow discourse parsing in Hindi and analyse the tasks of the identification of explicit discourse connectives and their arguments.

Further, we work on building a multilingual model for discourse unit segmentation and connective identification. Early approaches towards segmentation and connective detection relied on rule-based systems using POS tags and other syntactic information to identify discourse segments. Recently, transformer based neural systems have shown promising results in this domain. We establish a baseline using a bidirectional LSTM model. We then look at transformer based neural systems and train our model on 16 datasets encompassing 11 languages and 3 discourse annotation frameworks. This model gives state of the art performance for the English dataset. We then present a curated dataset and model for connective identification in...
Hindi. We experiment with different Indian language specific models and compare and analyse the model performance.
Contents

Chapter | Page
---|---
1 | Introduction
| 1.1 Coherence Relations | 2
| 1.1.1 Rhetorical Structure Theory Framework | 2
| 1.1.2 Segmented Discourse Representation Theory Framework | 3
| 1.1.3 Penn Discourse TreeBank Framework | 3
| 1.2 Discourse Unit Segmentation | 3
| 1.3 Connective Identification | 4
| 1.4 Motivation | 5
| 1.5 Key Contributions | 6
| 1.6 Thesis Overview | 7
2 | Related Work | 8
| 2.1 Discourse Unit Segmentation (RST/SDRT parsing) | 8
| 2.2 Explicit Connective Identification (PDTB parsing) | 9
3 | Approaches Towards Shallow Discourse Parsing | 11
| 3.1 Introduction | 11
| 3.2 Shallow Discourse Parsing in English | 12
| 3.2.1 Penn Discourse TreeBank | 12
| 3.2.2 Connective Identification | 13
| 3.2.3 Argument Extraction | 15
| 3.2.4 Relation Sense Classification | 16
| 3.3 Identification of Explicit Discourse Connectives and their Arguments in Hindi | 18
| 3.3.1 Hindi Discourse Relation Bank | 19
| 3.3.2 Disambiguation of Discourse Connectives | 19
| 3.3.3 Argument Extraction | 20
| 3.4 Summary | 20
4 | Multilingual Discourse Unit Segmentation and Connective Detection | 22
| 4.1 Introduction | 22
| 4.1.1 A focus on a uniform multilingual approach | 23
| 4.2 Data | 23
| 4.2.1 Annotation frameworks | 24
| 4.2.2 Languages | 24
| 4.3 System Overview | 25
CONTENTS

4.3.1 Bidirectional LSTM .......................................................................................................................... 25
4.3.2 BERT ................................................................................................................................................... 26
4.3.3 Our novel approach casting the problem as a token classification task .............................................. 27
  4.3.3.1 Handling out of vocabulary words and long input sequences .................................................. 28

4.4 Experimental Settings ............................................................................................................................ 29

4.5 Results .................................................................................................................................................... 29
  4.5.1 Bidirectional LSTM ........................................................................................................................ 29
  4.5.2 Transformer based model ............................................................................................................... 30

4.6 Conclusion ............................................................................................................................................... 31

5 Identification of Discourse Connectives in Hindi .................................................................................. 34
  5.1 Introduction ......................................................................................................................................... 34

5.2 Data Creation ....................................................................................................................................... 34
  5.2.1 Sourcing the data from the HDRB ................................................................................................. 35
  5.2.2 The Shakti Standard Format (SSF) .............................................................................................. 36
  5.2.3 Data Annotation ............................................................................................................................ 38

5.3 Approach and Methodology .................................................................................................................. 43
  5.3.1 Multilingual Representations for Indian Language (MuRIL) ......................................................... 44
  5.3.2 IndicBERT ..................................................................................................................................... 44

5.4 Results .................................................................................................................................................... 45

5.5 Conclusion ............................................................................................................................................... 46

6 Conclusions and Future Work .................................................................................................................. 47

Bibliography .................................................................................................................................................. 50
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>RST Tree Structure</td>
</tr>
<tr>
<td>1.2</td>
<td>biLSTM CRF model for segmentation</td>
</tr>
<tr>
<td>1.3</td>
<td>PDTB sense labels</td>
</tr>
<tr>
<td>3.1</td>
<td>Example of PDTB discourse parsing. Here, the connective is explicitly identified as the word 'if'. Arg1 and Arg2 denote the two text spans connected by the connective.</td>
</tr>
<tr>
<td>3.2</td>
<td>The basic pipeline architecture of an end-to-end shallow discourse parser for English. Here SS = Same Sentence, PS = Previous Sentence. SS and PS denote the location of Arg1 and Arg2 with respect to the discourse connective.</td>
</tr>
<tr>
<td>3.3</td>
<td>Top ten polysemous connectives (Explicit)</td>
</tr>
<tr>
<td>3.4</td>
<td>HDRB sense classification</td>
</tr>
<tr>
<td>3.5</td>
<td>Argument Identification Pipeline. SS = Same Sentence, PS = Previous Sentence, VG = Verb Group. SS and PS signify the relative position of the arguments with respect to the connective.</td>
</tr>
<tr>
<td>4.1</td>
<td>Data snippet from the English PDTB dataset</td>
</tr>
<tr>
<td>4.2</td>
<td>Basic structure of a biLSTM model</td>
</tr>
<tr>
<td>4.3</td>
<td>Masked LM pre-training task for BERT</td>
</tr>
<tr>
<td>4.4</td>
<td>System architecture and data flow</td>
</tr>
<tr>
<td>5.1</td>
<td>Example of sentential relatives</td>
</tr>
<tr>
<td>5.2</td>
<td>Example of subordinators</td>
</tr>
<tr>
<td>5.3</td>
<td>Example of particles</td>
</tr>
<tr>
<td>5.4</td>
<td>The head section in SSF. Information regarding the source, date, etc. is represented in this block.</td>
</tr>
<tr>
<td>5.5</td>
<td>The body section in SSF. Each line represents a word in the sentence in 4 parts.</td>
</tr>
<tr>
<td>5.6</td>
<td>A sentence annotated using the BIO format</td>
</tr>
<tr>
<td>5.7</td>
<td>Problem of ambiguity in Hindi discourse connectives</td>
</tr>
<tr>
<td>5.8</td>
<td>Connectives occurring as pairs. Tags B and I refer to Seg=B-Conn and Seg=I-Conn respectively.</td>
</tr>
<tr>
<td>5.9</td>
<td>Example 1 shows an adverbial phrase whereas Example 2 shows a postpositional phrase as an explicit discourse connective.</td>
</tr>
<tr>
<td>5.10</td>
<td>Subordinators functioning as discourse connectives</td>
</tr>
<tr>
<td>5.11</td>
<td>Emphatic particle</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

5.12 Outline of our approach ........................................... 43
5.13 Translation Language Modeling [28] ............................. 44
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Distribution of discourse relations in PDTB 2.0</td>
<td>14</td>
</tr>
<tr>
<td>3.2 List of features used by the classifier for connective identification [25]</td>
<td>15</td>
</tr>
<tr>
<td>3.3 List of features used by the classifier for argument extraction [25]</td>
<td>16</td>
</tr>
<tr>
<td>3.4 List of features used for argument extraction in Hindi [18]</td>
<td>21</td>
</tr>
<tr>
<td>4.1 Statistics for the 16 datasets, the last two columns display the number of training sentences and documents respectively</td>
<td>25</td>
</tr>
<tr>
<td>4.2 Final scores for the connective detection task on the PDTB framework datasets. The best scores for each dataset have been marked in bold. The model performs the best on the English PDTB dataset with an F1 score of 91.16%.</td>
<td>30</td>
</tr>
<tr>
<td>4.3 Results for the baseline bidirectional LSTM model for the tasks of EDU segmentation and connective detection. The model performs the best on the Dutch RST dataset with an F1 score of 85.308%.</td>
<td>32</td>
</tr>
<tr>
<td>4.4 Final scores for the EDU segmentation task on the treebanked data. The best score has been marked in bold. The model performs the best on the English RST-DT dataset with an F1 score of 97.00% which is the new state-of-the-art score for English.</td>
<td>32</td>
</tr>
<tr>
<td>4.5 Final scores for the EDU segmentation task on the plain tokenized data. The best score has been marked in bold. The model performs the best on the English RST-DT corpus with an F1 score of 95.23%.</td>
<td>33</td>
</tr>
<tr>
<td>5.1 The tagset used to mark the presence of an explicit discourse connective.</td>
<td>38</td>
</tr>
<tr>
<td>5.2 The final dataset statistics for our Hindi dataset created for the task of explicit connective identification in Hindi.</td>
<td>42</td>
</tr>
<tr>
<td>5.3 The final model scores on our Hindi dataset for the task of explicit connective identification in Hindi.</td>
<td>45</td>
</tr>
<tr>
<td>5.4 Comparison of the scores obtained by the multilingual BERT model on the newly created Hindi dataset with the scores on the 3 datasets mentioned in Chapter 4.</td>
<td>46</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Natural human language is a vital part of humankind. It allows us to share our ideas and thoughts with each other and progress as a society. For most people, the ability to understand and use a natural language comes naturally. However, enabling a machine to do the same is a very complex task. Natural Language Processing (NLP) is a branch of computer science that deals with the interaction between computers and natural human language. It involves enabling a machine to understand the basic underlying linguistic principles of language such as syntax, morphology, semantics, and pragmatics. Broadly, there are two major tasks that NLP deals with - Natural Language Understanding (helping a computer understand the structure and meaning of human language) and Natural Language Generation (enabling computers to generate human language with meaning). Discourse parsing is an integral part of natural language understanding and deals with understanding the information flow and modeling the coherence of a natural language texts.

With the development of several large treebanks and text processing tools such as shallow discourse parsers for high resource languages, discourse parsing is a popular and well-researched field in NLP. These tools and datasets act as a building block and further help improve the performance of downstream natural language processing tasks like question answering, sentiment analysis, machine translation. Despite having a large number of speakers, Indian languages such as Hindi, Marathi, Telugu, etc. do not have as many tools and resources as there are for English, although, we have noticed an improvement over the last decade.

Discourse parsing as a whole includes identifying discourse structures and labeling discourse relations. In this thesis, we present a multilingual model for discourse unit segmentation and connective identification across 16 datasets encompassing 11 languages. We then identify the lack of a similar resource for Hindi and describe our efforts towards creating a curated dataset and model for the connective identification task for the Hindi language.
1.1 Coherence Relations

Discourse parsing is a sub-field of natural language processing which involves understanding the structure, information flow, and modeling the coherence of a given text. A discourse is defined as a coherent, structured group of sentences [21]. Coherence is what separates a discourse from a random jumbled collection of sentences. Some examples of a discourse include a news article, a book, this thesis, a movie script, etc.

Coherence relations refer to the relations that connect the text spans present in a discourse. Consider the following two sentences:

Example 1: Raj went to eat a burger. He enjoys playing with his cat.
Example 2: Raj went to eat a burger. He was feeling hungry.

In the above two examples, we can see that Example 2 is much more coherent than Example 1. This is because we can connect the two sentences in Example 2 with the second sentence giving us a possible reason for the first sentence. This is not the case for Example 1 where we can’t establish a clear connection between the two sentences. We will discuss three common frameworks used to model coherence relations.

1.1.1 Rhetorical Structure Theory Framework

Rhetorical Structure Theory [38] (RST) is one of the most common frameworks used to model discourse relations. RST looks to parse a given discourse into a hierarchical tree structure by defining relations between two spans of text, a nucleus and a satellite. The nucleus is central to the discourse and definable independently. The satellite gives extra information about the nucleus. RST relations are represented by a tree structure with the relation represented by an arrow from the satellite to the nucleus as shown in Figure 1.1.

![Figure 1.1 RST Tree Structure](image)
1.1.2 Segmented Discourse Representation Theory Framework

Segmented Discourse Representation Theory (SDRT) describes a dynamic semantic theory of discourse interpretation which uses rhetorical relations to model the semantics/pragmatics interface. SDRT also assumes a hierarchical tree structure as described in the Rhetorical Structure Theory but with less constraints (including multiple relations between units). In both frameworks, the leaves of the tree structure correspond to the basic chunks of text spans which are known as Elementary Discourse Units (EDUs) or discourse unit segments.

1.1.3 Penn Discourse TreeBank Framework

The Penn Discourse TreeBank (PDTB) is a large-scale resource of annotated discourse relations and their arguments over the 1 million word Wall Street Journal (WSJ) Corpus. Discourse relations in the PDTB are marked by a discourse connective as shown in Example 1. However, some discourse relations may not be explicitly marked by a discourse connective as is apparent in Example 2. In this case, the neighbouring sentences are annotated as arguments with the connective being implicit.

Example 1: *He went away on vacation for an entire week. However, his wife had to continue working from home.*

Example 2: *Yesterday, the school football team won the first game of their season quite comfortably. (implicit=however) Due to unforeseen circumstances, their coach was not present to celebrate with them.*

The text spans in italics represent Argument 1, the connective is underlined and the text spans in bold represent Argument 2. The Hindi Discourse Relation Bank adopts the lexically grounded approach of the Penn Discourse Treebank, and describes the classification of Hindi discourse connectives. We revisit these discourse relation frameworks in detail and discuss the respective discourse parsing tasks in Chapter 4. Now, we will look at the respective unit identification tasks and how they differ across the annotation frameworks.

1.2 Discourse Unit Segmentation

For the Rhetorical Structure Theory and the Segmented Discourse Representation Theory frameworks, the task of discourse parsing corresponds to identifying the boundaries of the text spans known as elementary discourse units or EDUs. This is called discourse unit segmentation. An EDU can function as either a nucleus or a satellite. Both RST and SDRT frameworks aim to represent a discourse in a tree structure with a hierarchical arrangement of EDUs.
makes the task of EDU segmentation central to the discourse analysis of data which uses the RST and SDRT framework.

**Sentence 1**: Raj said that his cricket bat weighing 2 kilograms is worth Rs. 2000.

**Segmented form**: [Raj said]₁ [that his cricket bat]₂ [weighing 2 kilograms]₃ [is worth Rs. 2000.]₄

The above sentence consists of 4 EDUs which combine to form the complete sentence.

![Figure 1.2 biLSTM CRF model for segmentation](image)

Early approaches towards discourse unit segmentation relied on rule-based models leveraging syntactic information to identify the boundaries of EDUs. Modern systems employ machine learning based approaches leveraging the use of sequential neural models such as bidirectional LSTMs as shown in Figure 1.2. We will explain this task further in detail in Chapter 4.

### 1.3 Connective Identification

In contrast to the Rhetorical Structure Theory and the Segmented Discourse Representation Theory frameworks, the Penn Discourse TreeBank framework does not deal with building full tree structures as is the case in RST and SDRT, but instead focuses on the relations between text spans. This is why discourse parsing for datasets which follow the PDTB style framework is also typically referred to as shallow discourse parsing. The unit identification task for this framework is the identification of explicit discourse connectives. We will discuss this in further detail in Chapter 3.
PDTB discourse parsing consists of 3 basic tasks:

1. Identification of the discourse connectives.
2. Identification of the two text spans (arguments) of the discourse connectives.
3. Identification of the discourse relation type between the two arguments.

Apart from the argument structure of discourse relations, the PDTB framework also provides a sense label for the relations along with a hierarchical classification scheme as shown in Figure 1.3. This makes the PDTB framework useful for word sense disambiguation tasks. Early approaches towards connective identification make use of syntactic information to explicitly signal the presence of a discourse relation[46]. End to end shallow discourse parsers have also been built for the PDTB dataset[34].

![PDTB sense labels](image)

**Figure 1.3** PDTB sense labels[48]

### 1.4 Motivation

Discourse parsing forms the basis of several downstream NLP tasks such as text summarization, machine translation, question answering, sentiment analysis, etc. Discourse unit segmentation and connective identification are the fundamental tasks in discourse parsing. Improvements in the performance of the tools for these fundamental tasks results in better performance and
improved tools for all the aforementioned NLP problems. Discourse analysis methods are slowly improving and starting to become as reliable as the resources available for other NLP problems like parsing and syntactic tagging. This shows that leveraging new and improved techniques to build tools for these tasks and analysing their performance is the way forward. We present a novel approach towards these unit identification tasks in discourse parsing and report a new state of the art score in English.

Hindi is one the 22 official languages of India and is spoken by over 600 million people. The Hindi Discourse Relations Bank is aimed at developing a large corpus annotated with discourse relations. However, there is no proper dataset available for the task of connective identification in Hindi in a multilingual setting. In this thesis, we also present a curated dataset and model for this task in the Hindi language which will help add to the existing resources and also help in making the Hindi language available as a dataset for multilingual discourse parsing. Further benefits also include the availability of Hindi data for transfer learning development of cross lingual NLP tools for discourse parsing.

1.5 Key Contributions

In this thesis, we explore the topic of discourse parsing. The key contributions are listed as follows -

- We look at the task of shallow discourse parsing and analyse the approaches presented for the identification of a discourse relation. We explore a variety of approaches based on different models of coherence relations.

- We also analyse and evaluate an approach towards the identification of explicit discourse connectives and their arguments in the Hindi language.

- We present a multilingual model for discourse unit segmentation and connective identification across 16 datasets encompassing 11 languages.

- We present a novel approach for both these tasks and achieve state of the art scores for discourse unit segmentation on the English RST dataset along with improving the results across other languages as well.

- Noting the lack of a proper dataset for connective identification in Hindi, we present a curated dataset in Hindi for this task, building upon the data available as part of the Hindi Discourse Relation Bank (HDRB). Our dataset consists of 41,615 tokens over 1,850 sentences.
• We present a model for the automated identification of explicit discourse connectives in Hindi and also experiment using various Indian language specific models, comparing them with the performance of our above-mentioned multilingual model.

1.6 Thesis Overview

This thesis is presented in six chapters as follows:

Chapter 2: In this chapter, we look at the previous work done in the field of discourse parsing, early efforts employing rule-based techniques with syntactic information, recent improvements towards segmentation and connective detection.

Chapter 3: This chapter talks about the different approaches towards shallow discourse parsing and the major tasks involved in detecting the presence of a discourse relation. We compare and analyse the different approaches for each task and discuss the issues presented. We also look at an approach towards the identification of explicit discourse connectives and their arguments in the Hindi language.

Chapter 4: In this chapter, we present a multilingual model for discourse unit segmentation and connective identification. We evaluate the model for both tasks and present the results for each different dataset and annotation schema.

Chapter 5: We present the curated dataset and model for the connective identification task for the Hindi language in this chapter. We describe the SSF and CoNLL representations of Hindi data and discuss the BIO schema used for annotation of discourse connectives. We analyse and evaluate the model and report our results.

Chapter 6: In this chapter, we conclude by presenting a brief summary of the topics discussed in the thesis and discussing the directions for possible future work.
2.1 Discourse Unit Segmentation (RST/SDRT parsing)

Early work in the domain of discourse unit segmentation involved rule-based models which used syntactic information for the prediction of discourse segments and connectives [58]. The syntax of each sentence is used to insert the segment boundaries with further refinement done using lexical information obtained from the text. Le Thanh et al. [30] used syntactic information and cue phrases to segment sentences into EDUs and integrated constraints about textual adjacency and textual organization in a beam search for text level segmentation.

Fisher and Roark [13] investigated the approach towards segmentation using sentence level syntactic parse trees on the English RST-DT corpus. Soricut and Marcu [53] used probabilistic models for elementary discourse unit identification on the English RST-DT corpus [8]. They propose a two part solution - a statistical model is used to assign a probability of there being a discourse boundary after a word and a segmenter is used to insert the discourse boundaries based on the probabilities calculated. Building upon this, Subba and Di Eugenio [57] developed a neural network model for discourse segmentation using the same English RST-DT dataset. This was one of the first works with a neural approach towards discourse unit segmentation. A token based classifier was introduced by Sporleder and Lapata [54] which used POS tags, syntactic chunks, and clause information as features for the segmentation task.

Recent studies have been moving towards machine learning based approaches with most involving the use of sequential neural network models. Li et al. [32] used a bidirectional recurrent neural network along with a pointer network to select text boundaries in the input sequence. This model does not required hand crafted features. An end-to-end neural segmenter was proposed by Wang et al. [63] based on the BiLSTM-CRF framework. Their model uses a restricted self-attention mechanism and does not rely on syntactic features. Their system reported the then new state-of-the-art performance on the English RST-DT corpus with an F1
score of 94.3%. Braud et al. introduced the first multilingual segmenter across 5 languages and 3 non-newswire English domains using language independent tools.

Lukasik et al. investigate a transformers based approach towards document and discourse level segmentation with the English RST-DT corpus being used for discourse segmentation and the Wiki-727K dataset for document segmentation experiments. Koto et al. worked with an LSTM model and presented a top down approach towards RST discourse parsing by framing the task as a sequence labelling problem. Liu et al. worked on multilingual RST discourse parsing using a neural cross-lingual approach with multilingual vector representations and segment-level translation of the source content.

Discourse unit segmenters have also been developed for other languages. Lungen et al. proposed a discourse segmenter for German, adapting the segmentation principles used for English while considering the syntax, morphology, and the aspects of document structure of complex documents such as scientific articles. Iruskieta and Zapirain proposed a dependency based elementary discourse unit segmenter for the Basque language based on syntactic dependencies and linguistic rules. Afantenos et al. developed a segmenter for French building upon the standard classification techniques and adding a repairing heuristic. For Dutch, Nynke proposed a syntax based discourse segmentation model. Pardo et al. developed a cue phrase based analyser for scientific and news texts in Brazilian Portuguese. Da Cunha et al. proposed a segmenter for Spanish which uses lexical and syntactic rules and is based on the syntactic parser Freeling. Yang et al. presented a Chinese discourse segmenter using discourse commonality between English and Chinese, leveraging common features from bilingual data and giving a better performance than the baseline models using only a small amount of Chinese labeled data.

2.2 Explicit Connective Identification (PDTB parsing)

When it comes to the Penn Discourse TreeBank framework, since its release, plenty of research has been carried out for the subtasks involved in PDTB parsing. These include models for the disambiguation of discourse connectives, argument identification of discourse connectives as well as assigning sense tags to these relations. Miltsakaki et al. worked on sense disambiguation on the PDTB using syntactic features and a basic maximum entropy model. Building upon this, Pitler and Nenkova used syntactic features to help with the disambiguation of both - the discourse and non-discourse usage of a given word as well as sense disambiguation for the type of relation marked by a given connective.
One of the first works on argument identification using the PDTB was done by Elwell and Baldridge [12]. Prasad et al. [49] took this forward employing a sentence based representation of arguments and distinguishing between intra and inter-sentential connectives.

Sha and Pereira [51] presented one of the first approaches towards shallow parsing using conditional random fields. Early approaches towards shallow discourse parsing such as Ghosh et al. [14] employed the use of the aforementioned conditional random fields along with syntactic, semantic and lexical features obtained from the PDTB. The first end-to-end PDTB style discourse parser was developed by Lin et al. [34] using a fully data driven approach. The parser identifies all discourse connectives, their arguments, and assigns the relation sense labels to the relations between arguments. Building upon this, Wang et al. [62] described a refined PDTB parser capable of handling all the three tasks and following a pipeline architecture which then became the norm for all future end-to-end shallow discourse parsers.

The current state of the art end to end shallow discourse parser for English has been developed by Oepen et al. [41] which employs a similar pipeline architecture with a binary classifier for identifying explicit connectives, ranking of syntactic constituents for argument identification and two groups of classifiers for assigning the sense labels - one group to predict the senses of explicit relations and the other to assign the sense to non-explicit relations. Knaebel and Stede [24] studied the disambiguation of connectives for explicit discourse relations using contextualized word embeddings.

In the case of Hindi, the Hindi Discourse Relation Bank [42] was a massive step forward towards the development of tools for discourse parsing in Hindi. The first approach towards the identification of discourse connectives in Hindi was put forward by Jain et al. [19] where they proposed a maximum entropy classifier along with a curated feature set consisting of lexical and dependency information for the disambiguation of discourse connectives in Hindi. Further work towards discourse parsing in Hindi includes an approach towards the argument extraction of explicit discourse connectives by Jain et al. [18] where they adopted the pipeline approach introduced by Wang et al. [62] for shallow discourse parsing in English and added sub-tree extraction and linear tagging for extracting the arguments.
Chapter 3

Approaches Towards Shallow Discourse Parsing

3.1 Introduction

Given a large chunk of text with several sentences, the task of automatically determining the discourse relations between them is called discourse parsing. There are three major frameworks used to model discourse relations as mentioned before - Rhetorical Structure Theory (RST), Segmented Discourse Representation Theory (SDRT) and Penn Discourse TreeBank (PDTB). Based on the model of discourse relations used and the respective corpora, the task of discourse parsing differs across these frameworks.

In the case of the RST and SDRT frameworks, discourse parsing corresponds to the identification of EDUs and building RST parse trees. When it comes to the PDTB framework, the task is often called shallow discourse parsing and involves parsing the discourse into a set of discourse relations between two discourse units. The phrases 'PDTB discourse parsing' and 'Shallow discourse parsing' are synonymous. The word 'shallow' is added because unlike the RST and SDRT discourse parsing tasks, the PDTB task just deals with flat relations between text spans and does not involve connecting the discourse relations to obtain a tree structure. Figure 3.1 shows an example of PDTB discourse parsing.

In this chapter, we discuss the task of shallow discourse parsing for English. We describe the goal of the Penn Discourse TreeBank and look at the approaches towards the sub-tasks involved in shallow discourse parsing. We further go on to discuss the Hindi Discourse Relation Bank, a corpus aimed at providing a resource of annotated discourse relations in Hindi. We also discuss an approach towards the identification of arguments of explicit discourse connectives in Hindi.
3.2 Shallow Discourse Parsing in English

In this section, we describe the task of shallow discourse paring in English in detail. First, we look at the features of the Penn Discourse TreeBank, how it facilitates the development and evaluation of shallow discourse parsers and provides a large resource helpful in validating the many theories of discourse coherence structure.

A PDTB-style discourse parser detects and categorizes the discourse relations between discourse segments in a given text. There are three basic subtasks for PDTB discourse parsing as outlined by Wang et al.\cite{Wang2016}:

1: Connective Identification
2: Argument Extraction
3: Relation Sense Classification

We describe these sub-tasks involved in PDTB discourse parsing and talk about the approaches towards these sub-tasks.

3.2.1 Penn Discourse TreeBank

The Penn Discourse TreeBank follows a lexically grounded approach towards the annotation of discourse relations and provides a predicate-argument view for the same. The connective acts as a predicate and takes two text spans which act as its arguments. Arg2 refers to the text span which is syntactically attached to the connective and other text span is labelled as Arg1.
Figure 3.2 The basic pipeline architecture of an end-to-end shallow discourse parser for English. Here SS = Same Sentence, PS = Previous Sentence. SS and PS denote the location of Arg1 and Arg2 with respect to the discourse connective.

The PDTB defines two types of discourse relations depending on how they are realized in the discourse. The discourse relations where the connective is syntactically well-defined are called explicit discourse relations. In the absence of an explicit connective, the relations between two adjacent text spans are known as implicit discourse relations. In this case, the text span which comes first is labelled as Arg1 and the next one is labelled as Arg2. Apart from Explicit and Implicit discourse connectives, the PDTB defines 3 more labels. AltLex is for cases where using an implicit connective leads to redundancy caused by the alternative lexicalization of the relation due to some non-connective expression. EntRel refers to an entity-based relation perceived between two sentences and NoRel is for the instances where no relation is perceived between two sentences. Table 3.1 shows the distribution of relations in PDTB 2.0.

Now let us look at the three subtasks involved in shallow discourse parsing.

### 3.2.2 Connective Identification

The task of connective identification is the first step in PDTB discourse parsing. The presence of a discourse connective signifies that the relation between the two sentences is explicit. However, there may be cases where a discourse connective expression may not function as a
consider the following two sentences:

**Sentence 1:** Raj ate his food and he went outside to play.

**Sentence 2:** Raj likes to eat fruits and nuts.

In the above sentences, the word ‘and’ is present in both cases. However, it only acts as a discourse connective in Sentence 1, where it joins the two text spans ‘Raj ate his food’ and ‘he went outside to play’. Resolving this ambiguity is the main goal of any shallow discourse parser.

We will discuss two approaches towards this task. The first approach involves training a classifier using certain syntactic and lexical features for the disambiguation of the discourse connective expressions. The second approach presents connective identification as a sequence labeling task which uses sequence labeling models like CRF for identifying the discourse connectives.

The Penn Discourse TreeBank defines 100 different types of discourse connectives. In the classifier approach, the first step involves the extraction of all the possible connective expressions from the given text. These are then checked against the types mentioned in the PDTB for whether the expression functions as a discourse connective or not. For each occurrence of a discourse connective, several lexical and syntactic features are extracted and fed to the classifier. The classifier used is the Maximum Entropy text classifier. Table 3.2 shows the features used to train the classifier in this approach.

The second approach casts the task of connective identification as a sequence labeling task using Conditional Random Fields (CRFs). CRF is a probabilistic prediction model which assumes interdependency between features and considers future inputs as well while learning. The CRF model for this task trained using the following set of features to obtain the best performance:

1: The token itself
The POS tag of the token

The chunk tag, syntactic chuck which gives the location of the token in the BIO format

The BIO chain, path string of the syntactic tree nodes from the root node to the token.

<table>
<thead>
<tr>
<th>Lexical Features</th>
<th>Syntactic Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connective</td>
<td>Highest node in the parse tree</td>
</tr>
<tr>
<td>POS tag of connective</td>
<td>Context of the highest node</td>
</tr>
<tr>
<td>Connective and previous word</td>
<td>Presence of VP in the right sibling</td>
</tr>
<tr>
<td>Connective and next word</td>
<td>Path from the parent node of the connective to the root of the parse tree.</td>
</tr>
<tr>
<td>Location of connective in the sentence</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.2 List of features used by the classifier for connective identification

3.2.3 Argument Extraction

The second task in shallow discourse parsing is the identification of the two text spans which correspond to the Arg1 and Arg2 arguments of the discourse connective. In the case of implicit discourse relations, the task becomes pretty straightforward - for every pair of sentences which are adjacent to each other, if there is no explicit discourse relation present, assume the presence of an implicit discourse relation. There is the case where no discourse relation might be present, however, this is a very rare occurrence and is ignored.

The approaches towards this task are similar to the ones outlined for the first task of connective identification. In the case of explicit discourse connectives, Arg1 and Arg2 can either be in the same sentence or in different sentences. Consider the following two sentences:

**Sentence 1**: Raj ate his food and he went outside to play.

**Sentence 2**: Raj likes to eat fruits. However, he hates vegetables.

In Sentence 1, both arguments of the explicit discourse connective 'and' are present in the same sentence. However, in Sentence 2, the two text spans connected by the explicit connective 'However' reside in two different sentences. The task of argument extraction for explicit connectives then corresponds to finding the two text spans of the connective.

The two approaches towards this are similar to the ones discussed before. One is to use a binary classifier and check if a given text span functions as an argument. This is similar to the
first approach for connective identification, however, in this case, instead of a connective, we are dealing with sentences or clauses. To tackle this, it involves an extra step of identifying the pieces of the argument and then putting them together. The second approach again involves using sequence labeling models like CRFs to obtain the arguments of the connective.

In the first approach, constituents of the arguments are extracted from parse trees. After extracting the arguments, the classifier is used to determine the label for each argument candidate. The labels being Arg1, Arg2 and NULL. The same Maximum Entropy classifier is used. Table 3.3 shows the features used by the classifier for this task.

<table>
<thead>
<tr>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connective</td>
</tr>
<tr>
<td>Syntactic category of the connective</td>
</tr>
<tr>
<td>Sense class assigned to the connective in the PDTB corpus</td>
</tr>
<tr>
<td>Number of left/right siblings of the connective</td>
</tr>
<tr>
<td>Context of the constituent</td>
</tr>
<tr>
<td>Path from parent node of connective to the constituent node</td>
</tr>
<tr>
<td>Position of constituent relative to the connective</td>
</tr>
</tbody>
</table>

Table 3.3 List of features used by the classifier for argument extraction

Similar to connective identification, the second approach towards argument extraction again treats it as a sequence labeling task and uses CRFs with different sets of features to extract the arguments. Along with the four features mentioned before, four extra features are added to the feature set:

1: Dependency chain, similar to the BIO chain
2: VerbNet class
3: Output of the connective identification model
4: Output of the Arg2 span extraction model

3.2.4 Relation Sense Classification

The Penn Discourse TreeBank provides sense labels for Explicit, Implicit and AltLex relations. The third task in PDTB parsing corresponds to determining the sense of the discourse relation. However, a discourse connective can have more than one sense label depending upon
the content of the argument text spans and the context. In such cases, this task basically becomes the problem of word sense disambiguation.

<table>
<thead>
<tr>
<th>Connective</th>
<th>Senses</th>
</tr>
</thead>
<tbody>
<tr>
<td>after</td>
<td>succession (523), succession-reason (50), other (4)</td>
</tr>
<tr>
<td>since</td>
<td>reason (94), succession (78), succession-reason (10), other (2)</td>
</tr>
<tr>
<td>when</td>
<td>Synchrony (477), succession (157), general (100), succession-reason (65), Synchrony-general (50), Synchrony-reason (39), hypothetical (11), implicit assertion (11), Synchrony-hypothetical (10), other (69)</td>
</tr>
<tr>
<td>while</td>
<td>juxtaposition (182), Synchrony (154), Contrast (120), expectation (79), opposition (78), Conjunction (39), Synchrony-juxtaposition (26), Synchrony-Conjunction (21), Synchrony-Contrast(22), COMPARISON (18), Synchrony-opposition (11), other (31)</td>
</tr>
<tr>
<td>meanwhile</td>
<td>Synchrony-Conjunction (92), Synchrony (26), Conjunction (25), Synchrony-juxtaposition (15), other(35)</td>
</tr>
<tr>
<td>but</td>
<td>Contrast (1609), juxtaposition (636), contra-expectation (494), COMPARISON (260), opposition (174), Conjunction (63), Conjunction-Pragmatic contrast (14), Pragmatic-contrast (14), other (32)</td>
</tr>
<tr>
<td>however</td>
<td>Contrast (254), juxtaposition (89), contra-expectation (70), COMPARISON (49), opposition (31), other (12)</td>
</tr>
<tr>
<td>although</td>
<td>expectation (132), Contrast (114) juxtaposition (34), contra-expectation (21), COMPARISON (16), opposition (9), other (2)</td>
</tr>
<tr>
<td>and</td>
<td>Conjunction (2543), List (210), result-Conjunction (138), result (38), precedence-Conjunction (30), juxtaposition (11), other(30)</td>
</tr>
<tr>
<td>if</td>
<td>hypothetical (682), general (175), unreal present (122), factual present (73), unreal past (53), expectation (34), implicit assertion (29), relevance (20), other (31)</td>
</tr>
</tbody>
</table>

**Figure 3.3** Top ten polysemous connectives (Explicit)

In case of explicit connectives, the sense can be predicted with high accuracy using classification models like Maximum Entropy, SVM, etc. Sense classification for implicit discourse relations, however, needs information from the two text spans involved in the implicit relation. In this case, along with using the classification models with certain features such as VerbNet classes, Brown Clusters and syntactic parses, other approaches involving the use of neural models such as feed forward NNs, CNNs, and RNNs with word embeddings as inputs were also used.

The PDTB parsers in English are evaluated on all the three above-mentioned subtasks. As a result, even though the systems for the task of connective identification perform reasonably well on a blind test with F1 scores in the low 90s, the overall parser scores are dragged down due to the low scores for the tasks of argument extraction (especially Arg1) and relation sense classification. The current overall state of the art PDTB parser reports a final F1 score of 27.7% on strict matching.
3.3 Identification of Explicit Discourse Connectives and their Arguments in Hindi

In this section, we describe the Hindi Discourse Relation Bank in detail and discuss an approach towards the identification of explicit discourse connectives and their arguments in Hindi.

Figure 3.4 HDRB sense classification[42]
3.3.1 Hindi Discourse Relation Bank

The Hindi Discourse Relation Bank (HDRB) is a project that largely follows the approach of the Penn Discourse TreeBank and provides a resource for the characterization of discourse connectives, their arguments, and their senses in the Hindi language[42]. The HDRB corpus consists of 200K words and was annotated on a subset of the Hindi TreeBank[2]. The source texts are taken from the Hindi newspaper Amar Ujala and consist of news article from different domains such as sports, film, politics.

Similar to the PDTB approach, discourse relations are marked as either Explicit, Implicit, AltLex, EntRel or NoRel. The arguments of the discourse relation consist of the minimal text span required to interpret the relation and are marked as Arg1 and Arg2. Similar to the PDTB, relation senses are also assigned following a hierarchical sense classification as shown above in Figure 3.4.

Apart from having the same four top level sense classes - Temporal, Contingency, Comparison and Expansion - the HDRB provides further clarification at the second type and third subtype levels in order to capture additional senses and take care of some language-specific issues. These changes can be seen while comparing Figures 1.3 and 3.4. In the PDTB sense classification, the tags at the type level relay information about the relations’ semantics and the subtype level tags depict the ordering of the arguments. The HDRB removes the argument ordering tags from the subtype level. Thus, all levels in the HDRB are used for the refinement of the relations’ semantics but to varying levels of coherence.

The second major difference between the PDTB and HDRB sense labelling is the pragmatic senses. The HDRB replaces the PDTB pragmatic sense by classifying each pragmatic sense at the type level into three uniform subtypes - epistemic, speech-act and propositional [42]. A new type named ‘Goal’ has also been added in the Contingency class which is used when the situation in the first argument is a goal of the situation in the second argument.

3.3.2 Disambiguation of Discourse Connectives

Similar to the PDTB, the first step in Hindi discourse parsing is the identification of discourse connectives. In this section, the task objective is to distinguish between when a certain word functions as a discourse connective and when it doesn’t. Jain et al.[19] adopt a classifier based approached using specific features extracted from the Hindi data to improve classification performance.

The base lexical features used include the connective itself, the POS tag of the connective, the POS tags of the two neighbours of the connective (one in front, one behind), the chunk tag of the connective and its two neighbours. Along with these lexical features, using dependency relation, coordination between clauses, right word location, and a combination of all dependency
features make up the final feature set which yielded the current state of the art performance for Hindi with an F1 score of 90.8%.

3.3.3 Argument Extraction

This task follows the similar structure of subtasks as its English counterpart. Jain et al.\cite{18} propose a pipeline architecture similar to Lin et al.\cite{34}, using subtree extraction to get the Arg1 and Arg2 candidates followed by CRF tagging. Figure 3.5 shows the pipeline model used for this task.

![Argument Identification Pipeline](image)

**Figure 3.5** Argument Identification Pipeline\cite{18}. SS = Same Sentence, PS = Previous Sentence, VG = Verb Group. SS and PS signify the relative position of the arguments with respect to the connective.

Arg1 and Arg2 extraction is done using the Maximum Entropy classifier along with different feature sets followed by Conditional Random Field tagging for further refining the extent of the extracted arguments. Table 3.4 enlists all the features used for the classification tasks. The best final scores for Arg1 and Arg2 identification are 71.1% and 93.2% respectively.

3.4 Summary

We have presented the different approaches towards shallow discourse parsing in English and Hindi. We discussed about the Penn Discourse TreeBank which will be touched upon again in Chapter 4. We looked at the Hindi Discourse Relation Bank which will be used in Chapter 5. We now move on to the next chapters where we present the main contributions of this thesis in the sub-field of discourse parsing.
<table>
<thead>
<tr>
<th><strong>Feature Name</strong></th>
<th><strong>Feature Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Conn-Str</td>
<td>Connective Itself</td>
</tr>
<tr>
<td>Conn-Pos</td>
<td>POS tag of the connective</td>
</tr>
<tr>
<td>Conn-Pos Sentence</td>
<td>Position of the connective in the sentence (Start/Middle)</td>
</tr>
<tr>
<td>Is-Leaf-Node</td>
<td>Connective node is a leaf node in the dependency tree</td>
</tr>
<tr>
<td>Node-Tag</td>
<td>Chunk tag of the node</td>
</tr>
<tr>
<td>Verb-Group</td>
<td>Verb group &amp; POS tag</td>
</tr>
<tr>
<td>Verb-Group Compact</td>
<td>Verb group and POS tag consisting of main verb and auxiliary verbs</td>
</tr>
<tr>
<td>Verb-Root Inflection</td>
<td>Root and Inflection of the main and auxiliary verbs</td>
</tr>
<tr>
<td>VG-In SubTree</td>
<td>Presence of VG node in sub tree of node</td>
</tr>
<tr>
<td>VG-In-Parent SubTree</td>
<td>Presence of VG node in parent of node</td>
</tr>
<tr>
<td>Is-Conn</td>
<td>Node is a part of a discourse connective</td>
</tr>
<tr>
<td>Conn-Rel Pos</td>
<td>Position of chunk w.r.t connective in sentence</td>
</tr>
<tr>
<td>Clause-End</td>
<td>Presence of clause boundary</td>
</tr>
<tr>
<td>Right-Word Location</td>
<td>Location of the word following the connective in the dependency tree w.r.t connective node</td>
</tr>
<tr>
<td>Arg2-Pos</td>
<td>Position of Arg2 in dependency tree</td>
</tr>
<tr>
<td>Part-Conn SubTree</td>
<td>Indicates whether node is part of connective subtree</td>
</tr>
<tr>
<td>Chunk-Before Conn</td>
<td>Number of chunks before discourse connective</td>
</tr>
<tr>
<td>Conn-Two Clause</td>
<td>Indicates the presence of two verb groups as children to connective node.</td>
</tr>
</tbody>
</table>

**Table 3.4** List of features used for argument extraction in Hindi.
Chapter 4

Multilingual Discourse Unit Segmentation and Connective Detection

4.1 Introduction

Discourse segmentation is a fundamental task in discourse parsing, which involves identifying the minimal chunks of text that combine to form a coherent discourse. In this chapter, we present our efforts towards building a model for multilingual discourse unit segmentation and connective identification. We also describe the different datasets and discourse relation frameworks used for training and testing.

In the Rhetorical Structure Theory (RST)\cite{38} framework, the basic chunks of texts are known as Elementary Discourse Units (EDUs). These EDUs are linked together by discourse relations which may be explicit (when explicitly marked in the text by a discourse connective) or implicit. Segmentation refers to the task of identifying these EDUs. Another popular framework is the Segmented Discourse Representation Theory (SDRT)\cite{29}. Both these frameworks segment the text into non-overlapping spans covering entire documents. The discourse segmentation task, in this case, corresponds to identifying the starting point of each discourse unit. In 2008, the Penn Discourse TreeBank (PDTB)\cite{48} was released with a corpus of over 1 million words. The unit identification task for this framework corresponds to identifying the spans of discourse connectives that explicitly identify the existence of a discourse relation.

The task of EDU segmentation has been widely researched in the past due to its importance as a building block for further downstream natural language processing tasks. Most of the previous research concerning EDU segmentation has relied on the information obtained from syntactic elements of the text such as syntactic parse trees (\cite{58} ; \cite{31}). However, recent works have explored the task of segmentation using systems based on neural networks using the BiLSTM - CRF framework\cite{63} or the attention mechanism\cite{33}. 

22
4.1.1 A focus on a uniform multilingual approach

We focus on using language independent tools to develop a multilingual system capable of handling different annotation frameworks and language guidelines resulting in a flexible and robust model which still remains accurate enough to help with the development of further tools for other discourse parsing tasks. The multilingual setting also helps establish a common baseline for the annotation of discourse unit segments and connectives making it easier to add to the existing data and improve the current systems. This will become more apparent in the next chapter when we add to the datasets available by creating a new dataset for connective identification in the Hindi language.

4.2 Data

In this section, we describe the datasets used for training, testing and evaluating the model performance. The data consists of 16 datasets comprising of 11 languages (German, English, Basque, Persian, French, Dutch, Portuguese, Russian, Spanish, Turkish, and Mandarin Chinese). Each dataset consists of two formats - one is the treebanked format which provides the gold syntax and sentence boundaries for each sentence and the other is the plain tokenized data which consists of entire documents without the gold syntax. This encourages the system design to be able to adapt and deal with data when the gold syntax is not provided.

![Figure 4.1 Data snippet from the English PDTB dataset](image-url)
4.2.1 Annotation frameworks

The data follows three annotation frameworks - RST, SDRT, and PDTB. The tasks differ across different formalisms. In the RST and SDRT framework, the text has been segmented into non-overlapping spans covering each entire documents. The corresponding task, in this case, is finding the starting point of each discourse unit. In the PDTB framework, the segmentation task corresponds to identifying the spans of discourse connectives that explicitly identify the presence of a discourse relation.

Out of the 16 datasets, 3 corpora follow the PDTB framework (English, Turkish, and Mandarin Chinese), 2 are represented by the SDRT framework (English and French), and 11 datasets follow the RST framework. The diversity across the datasets with respect to the frameworks encourages the design of flexible systems capable of dealing with multiple formalisms.

The data follows the CoNLL-U format with each row depicting a token (word) and consisting of 10 fields, the first field being the word index starting from 1 for each new sentence (in the case of treebanked data) or each new document (in the case of plain tokenized data). The label denoting the presence of a discourse connective or a discourse segment boundary is marked in the 10th column of the row. Figure 4.1 shows a snippet of the dataset taken from the English PDTB dataset used in this section.

4.2.2 Languages

The 16 datasets consist of 11 languages distributed as follows:

- **English**: 4 datasets ([48], [65], [8], [1])
- **Spanish**: 2 datasets ([10], [6])
- **Mandarin Chinese**: 2 datasets ([67], [6])
- **German**: 1 dataset ([55])
- **French**: 1 dataset ([45])
- **Russian**: 1 dataset ([59])
- **Portuguese**: 1 dataset ([7])
- **Dutch**: 1 dataset ([50])
- **Basque**: 1 dataset ([16])
- **Turkish**: 1 dataset ([66])
- **Persian**: 1 dataset ([52])

We present a comprehensive overview and statistics of all the datasets in Table 4.1. The English PDTB corpus is the largest in the given datasets with 1,061,229 training tokens and 1,156,657 total tokens in the dataset. The Chinese section of the RST Spanish - Chinese treebank is the smallest corpus with 9,655 training tokens and a total of 15,496 tokens.
<table>
<thead>
<tr>
<th>Corpus</th>
<th>Language</th>
<th>Framework</th>
<th>Train Tokens</th>
<th>Train Sent.</th>
<th>Train Docs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>deu.rst.pcc</td>
<td>German</td>
<td>RST</td>
<td>26,831</td>
<td>1,773</td>
<td>142</td>
</tr>
<tr>
<td>eng.rst.gum</td>
<td>English</td>
<td>RST</td>
<td>116,557</td>
<td>6,346</td>
<td>128</td>
</tr>
<tr>
<td>eng.rst.rstdt</td>
<td>English</td>
<td>RST</td>
<td>166,854</td>
<td>6,672</td>
<td>309</td>
</tr>
<tr>
<td>eus.rst.ert</td>
<td>Basque</td>
<td>RST</td>
<td>30,690</td>
<td>1,599</td>
<td>116</td>
</tr>
<tr>
<td>fas.rst.prstc</td>
<td>Persian</td>
<td>RST</td>
<td>52,497</td>
<td>1,713</td>
<td>120</td>
</tr>
<tr>
<td>nld.rst.nlhd</td>
<td>Dutch</td>
<td>RST</td>
<td>17,562</td>
<td>1,156</td>
<td>56</td>
</tr>
<tr>
<td>por.rst.cstn</td>
<td>Portuguese</td>
<td>RST</td>
<td>52,177</td>
<td>1,825</td>
<td>114</td>
</tr>
<tr>
<td>rus.rst.rrt</td>
<td>Russian</td>
<td>RST</td>
<td>390,375</td>
<td>18,932</td>
<td>272</td>
</tr>
<tr>
<td>spa.rst.rststb</td>
<td>Spanish</td>
<td>RST</td>
<td>43,055</td>
<td>1,548</td>
<td>203</td>
</tr>
<tr>
<td>spa.rst.sctb</td>
<td>Spanish</td>
<td>RST</td>
<td>10,253</td>
<td>326</td>
<td>32</td>
</tr>
<tr>
<td>zho.rst.sctb</td>
<td>Chinese</td>
<td>RST</td>
<td>9,655</td>
<td>361</td>
<td>32</td>
</tr>
<tr>
<td>eng.sdrt.stac</td>
<td>English</td>
<td>SDRT</td>
<td>41,060</td>
<td>8,754</td>
<td>33</td>
</tr>
<tr>
<td>fra.sdrt.annodis</td>
<td>French</td>
<td>SDRT</td>
<td>22,515</td>
<td>1,020</td>
<td>64</td>
</tr>
<tr>
<td>eng.pdtb.pdtb</td>
<td>English</td>
<td>PDTB</td>
<td>1,061,229</td>
<td>44,563</td>
<td>1,992</td>
</tr>
<tr>
<td>tur.pdtb.tdb</td>
<td>Turkish</td>
<td>PDTB</td>
<td>398,515</td>
<td>24,960</td>
<td>159</td>
</tr>
<tr>
<td>zho.pdtb.cdtb</td>
<td>Chinese</td>
<td>PDTB</td>
<td>52,061</td>
<td>2,049</td>
<td>125</td>
</tr>
</tbody>
</table>

Table 4.1 Statistics for the 16 datasets, the last two columns display the number of training sentences and documents respectively.

4.3 System Overview

In this section, we describe our approach towards the tasks of discourse unit segmentation and connective identification. We establish a baseline using a bidirectional LSTM classifier and propose a final system leveraging the transformers architecture\(^1\) with a classification layer on top.

4.3.1 Bidirectional LSTM

Bidirectional LSTMs are basically an extension to the LSTM model\(^1\) and can be thought of as joining two LSTMs together. This architecture allows the model to read input sequences in both forward and backward directions, which results in an effective capture of context, which helps in improving model performance for classification tasks. Figure 4.2 shows the basic structure of a bidirectional LSTM model.

As a baseline, we propose a PyTorch\(^2\) implementation of a simple bidirectional LSTM which encodes input tokens using word embeddings and has a single linear layer for the segmentation task. The model takes in an input sequence of tokens along with the corresponding sequence of labels, converts them into word vectors, and passes them to the biLSTM. The final linear layer

\(^1\)https://pytorch.org/
Figure 4.2 Basic structure of a biLSTM model

takes the hidden states as input from the bidirectional LSTM and outputs the final segment boundaries.

4.3.2 BERT

Recent studies have shown promising results with neural approaches towards NLP tasks. Transformer based architectures like BERT achieved state-of-the-art scores on several natural language processing tasks. Following this, Devlin et al. proposed a multilingual BERT model pretrained on 104 languages in a self-supervised fashion. We adopt the aforementioned model, modify it and finetune it for the segmentation and connective detection tasks.

BERT is a language representation model developed by researchers at Google AI language. BERT uses a multi-layer bidirectional Transformer encoder, the attention mechanism described by Vaswani et al., to gain contextual information between words in a text.

Pretraining of BERT employs two training strategies. The first task is called Masked LM. In this task, 15% of input tokens are randomly selected and masked. BERT then attempts to only predict those masked tokens instead of reconstructing the full input sequence. Figure 4.3 shows the Masked LM task. The second task is called Next Sentence Prediction (NSP).
This focuses on training the model to understand relationships between sentences. In this task, the training input to BERT consists of pairs of sentences. In this input, 50% of the pairs are sentences where Sentence 2 is the actual next sentence that follows Sentence 1. The rest of the times it is a random sentence from the training dataset.

4.3.3 Our novel approach casting the problem as a token classification task

Previous approaches towards EDU segmentation adopt a sequence classification approach. Wang et al. [63] employed a self-attention mechanism restricting the attention area for the model to a neighborhood of fixed size which would prevent the unnecessary tokens from misleading the model. Building upon this, we present the task of segmentation as a token classification problem to the multilingual BERT model. This enables the architecture to learn and gain information from each individual token and spares it from committing errors due to unnecessary tokens. The multilingual BERT model has been trained on a huge corpus of multilingual data which makes it capable of being fine-tuned to handle downstream tasks such as token classification even when the available training dataset is not very large.

Figure 4.4 illustrates the basic architecture and data flow in our proposed system. The results obtained from our system indicate that fine-tuning the multilingual BERT model by
considering discourse unit segmentation as a token classification task instead of a sequence labeling task leads to better model performance.

Figure 4.4 System architecture and data flow

4.3.3.1 Handling out of vocabulary words and long input sequences

Our system takes sequences of tokens and corresponding labels as input which are then fed into a WordPiece tokenizer. The WordPiece tokenizer converts the words into tokens to be fed into the model. One issue that arises, in this case, is the problem of out of vocabulary words. In such cases, the tokenizer splits the input word into subtokens which are present in the tokenizer vocabulary. This creates a mismatch between the number of input tokens and labels. We handle this issue by modifying the label tokenization process to match the output length. The tokenizer returns an offset mapping for each split word, according to which we then modify the label encodings. This ensures efficiency in the training process. We obtain the segmentation outputs by adding a linear classification layer that takes the outputs from the hidden states and converts them into the segmentation boundary labels.

This system is suited for the treebanked data since the lengths of the input sequences stay below the threshold of 512 tokens. However, in the case of plain tokenized documents, the input sequences are entire documents whose length often goes beyond the 512 token limit. We work around this by adopting the approach of Muller et al. [40] and using the StanfordNLP sentence splitter for splitting the input document sequences into sentences.

https://stanfordnlp.github.io/CoreNLP/ssplit.html
4.4 Experimental Settings

For the bidirectional LSTM model, we used 300 dimensional randomly initialized word embeddings to encode the input tokens and pass it to a hidden layer with 100 dimensions and a dropout rate of 0.5. We used the Adam optimizer to update the model weights with a learning rate of 1e-3 and a batch size of 1 for training. We used the negative loglikelihood loss function and trained the model for 5 epochs to obtain the optimal results.

For our final system, we used HuggingFace’s PyTorch implementation of the multilingual BERT model. We fine tuned the model to work with the discourse unit segmentation task and present the optimal model settings for the same. We used a batch size of 8 for the training of the model. We experimented with various preprocessing and training parameters, changing the padding and truncation length, tweaking the weight update rate, and changing the number of training epochs. We found that the best results were obtained when the input sequences in the batch were padded to the length of the longest sequence in that given batch. We used the AdamW optimizer with a learning rate of 1e-5 and trained for 5 epochs to obtain the best F1 scores.

4.5 Results

Our final system improves the previous best results for 15 out of the 16 datasets for the discourse unit segmentation and the connective identification tasks. In this section, we present the detailed results of all our systems. The final precision, recall and F1 scores have been noted by averaging the scores obtained over 5 runs.

4.5.1 Bidirectional LSTM

We report the baseline accuracy, precision, recall, and F1 scores on the development and test sets for the datasets based on the outputs obtained from the bidirectional LSTM model in Table 4.3. The bidirectional LSTM model gives largely decent scores on most of the datasets, with the lowest being 65.979% F1 score for the segmentation task on the zho.rst.sctb dataset and 66.292% F1 score for the connective identification task on the tur.pdtb.tdb dataset. For the segmentation task, the model attains its best performance on the nld.rst.nldt dataset with an F1 score of 85.362% and similarly, 78.629% F1 score on the eng.pdtb.pdtb dataset for connective identification task.

The low F1 scores on the Chinese RST dataset can be attributed to the small training dataset size with only 32 documents consisting of 9,655 tokens available for training. On many
Table 4.2  Final scores for the connective detection task on the PDTB framework datasets. The best scores for each dataset have been marked in bold. The model performs the best on the English PDTB dataset with an F1 score of 91.16%.

datasets, the precision scores are quite higher than the recall (Basque dataset - 95% precision and 61% recall, Russian dataset - 84% precision and 60% recall), indicating that the model is primarily aiming for the generic discourse unit boundary detection at the beginning of the discourse segments.

4.5.2 Transformer based model

We report the discourse segmentation results from our multilingual transformers based model in Tables 4.4 and 4.5. Our system returns high F1 scores across all 16 provided datasets and manages to improve the previous best scores on all except the French ANNODIS corpus in the treebanked scenario. We also report better scores for 14 out of the 16 datasets in the plain document level setting with only the French ANNODIS corpus and the Chinese section of the RST Spanish-Chinese Treebank, yielding scores lower than the previous best results.

We also report the first ever scores for the Persian RST corpus\footnote{52} with our system returning the best F1 score of 92.32% in the treebanked scenario with the gold syntax and the best F1 score of 91.90% on the plain tokenized documents.

We observe significant improvements compared to the previous results on the Basque dataset (F1 score of 88.52% compared to the previous best F1 score of 84.06%) with the plain tokenized documents. To the best of our knowledge, we also report state-of-the-art performance so far for discourse unit segmentation on the English RST-DT dataset with an F1 score of 97.09% on the test dataset with the gold syntax.

With respect to the task of connective identification on the datasets with the PDTB framework, we report the F1 scores for our model on these datasets in Table 4.2 for the treebanked and plain tokenized scenario.
Comparing the results of the model on the treebanked data with the results on the plain tokenized documents, we can see that the performance of the model is consistently better on the treebanked datasets, with the Chinese section of the RST Spanish-Chinese Treebank showing the most significant drop in F1 scores (81.29% to 71.70%). There are a couple of exceptions, with the model giving better scores for the Basque and Spanish RST datasets. However, in general, we can conclude that the presence of gold sentence boundaries helps the system perform better compared to the datasets with predicted sentence boundary markers.

4.6 Conclusion

In this chapter, we discuss a novel approach towards building a system for the EDU segmentation and connective identification tasks across 16 datasets consisting of 11 languages and 3 frameworks. We present a multilingual system leveraging the new transformer based architecture and fine-tune it to obtain quality results for the two above-mentioned tasks. Our results show that a token classification approach towards the task of segmentation helps the model perform better and results in state-of-the-art scores. Automated discourse analysis methods are slowly becoming as reliable as the resources available for syntactic parsing. Increasing the data resources by adding new language datasets and analysing their discourse relations will be a big step towards achieving this goal. We will discuss our contributions towards this in the next chapter.
<table>
<thead>
<tr>
<th>Corpus</th>
<th>Development Set</th>
<th>Test Set</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc</td>
<td>P</td>
</tr>
<tr>
<td>deu.rst.pcc</td>
<td>97.239</td>
<td>89.166</td>
</tr>
<tr>
<td>eng.pdtb.pdtb</td>
<td>98.873</td>
<td>81.446</td>
</tr>
<tr>
<td>eng.rst.gum</td>
<td>95.226</td>
<td>80.000</td>
</tr>
<tr>
<td>eng.rst.rstdt</td>
<td>96.614</td>
<td>82.961</td>
</tr>
<tr>
<td>eng.sdrt.stac</td>
<td>90.878</td>
<td>79.049</td>
</tr>
<tr>
<td>eus.rst.ert</td>
<td>95.927</td>
<td>93.821</td>
</tr>
<tr>
<td>fas.rst.prstc</td>
<td>97.934</td>
<td>90.380</td>
</tr>
<tr>
<td>fra.sdrt.annodis</td>
<td>96.120</td>
<td>91.666</td>
</tr>
<tr>
<td>nld.rst.nlnt</td>
<td>97.488</td>
<td>90.522</td>
</tr>
<tr>
<td>por.rst.cstn</td>
<td>95.974</td>
<td>75.698</td>
</tr>
<tr>
<td>rus.rst.rrt</td>
<td>95.892</td>
<td>84.981</td>
</tr>
<tr>
<td>spa.rst.rststb</td>
<td>98.119</td>
<td>84.367</td>
</tr>
<tr>
<td>spa.rst.sctb</td>
<td>97.834</td>
<td>79.761</td>
</tr>
<tr>
<td>tur.pdtb.tdb</td>
<td>98.741</td>
<td>63.533</td>
</tr>
<tr>
<td>zho.rst.sctb</td>
<td>97.040</td>
<td>68.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Corpus</th>
<th>Test Set</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc</td>
<td>P</td>
</tr>
<tr>
<td>deu.rst.pcc</td>
<td>97.190</td>
<td>90.763</td>
</tr>
<tr>
<td>eng.pdtb.pdtb</td>
<td>98.882</td>
<td>78.870</td>
</tr>
<tr>
<td>eng.rst.gum</td>
<td>95.247</td>
<td>78.860</td>
</tr>
<tr>
<td>eng.rst.rstdt</td>
<td>96.436</td>
<td>81.785</td>
</tr>
<tr>
<td>eng.sdrt.stac</td>
<td>90.835</td>
<td>79.246</td>
</tr>
<tr>
<td>eus.rst.ert</td>
<td>96.112</td>
<td>95.974</td>
</tr>
<tr>
<td>fas.rst.prstc</td>
<td>97.272</td>
<td>90.500</td>
</tr>
<tr>
<td>fra.sdrt.annodis</td>
<td>95.678</td>
<td>90.890</td>
</tr>
<tr>
<td>nld.rst.nlnt</td>
<td>97.382</td>
<td>91.525</td>
</tr>
<tr>
<td>por.rst.cstn</td>
<td>96.749</td>
<td>74.863</td>
</tr>
<tr>
<td>rus.rst.rrt</td>
<td>95.734</td>
<td>84.391</td>
</tr>
<tr>
<td>spa.rst.rststb</td>
<td>97.756</td>
<td>80.616</td>
</tr>
<tr>
<td>spa.rst.sctb</td>
<td>97.771</td>
<td>81.679</td>
</tr>
<tr>
<td>tur.pdtb.tdb</td>
<td>98.693</td>
<td>62.632</td>
</tr>
<tr>
<td>zho.rst.sctb</td>
<td>97.232</td>
<td>78.048</td>
</tr>
</tbody>
</table>

Table 4.3 Results for the baseline bidirectional LSTM model for the tasks of EDU segmentation and connective detection. The model performs the best on the Dutch RST dataset with an F1 score of 85.308%.

<table>
<thead>
<tr>
<th>Corpus</th>
<th>P</th>
<th>R</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>deu.rst.pcc</td>
<td>93.71</td>
<td>96.26</td>
<td>94.97</td>
</tr>
<tr>
<td>eng.rst.gum</td>
<td>91.47</td>
<td>95.67</td>
<td>93.52</td>
</tr>
<tr>
<td>eng.rst.rstdt</td>
<td>96.15</td>
<td>98.04</td>
<td><strong>97.09</strong></td>
</tr>
<tr>
<td>eng.sdrt.stac</td>
<td>96.73</td>
<td>95.67</td>
<td>96.20</td>
</tr>
<tr>
<td>eus.rst.ert</td>
<td>87.23</td>
<td>88.65</td>
<td>87.94</td>
</tr>
<tr>
<td>fas.rst.prstc</td>
<td>90.53</td>
<td>94.18</td>
<td>92.32</td>
</tr>
<tr>
<td>fra.sdrt.annodis</td>
<td>86.76</td>
<td>88.03</td>
<td>87.39</td>
</tr>
<tr>
<td>nld.rst.nlnt</td>
<td>98.16</td>
<td>94.67</td>
<td>96.39</td>
</tr>
<tr>
<td>por.rst.cstn</td>
<td>92.60</td>
<td>94.12</td>
<td>93.35</td>
</tr>
<tr>
<td>rus.rst.rrt</td>
<td>84.93</td>
<td>85.18</td>
<td>85.06</td>
</tr>
<tr>
<td>spa.rst.rststb</td>
<td>89.12</td>
<td>94.35</td>
<td>91.66</td>
</tr>
<tr>
<td>spa.rst.sctb</td>
<td>92.31</td>
<td>78.57</td>
<td>84.89</td>
</tr>
<tr>
<td>zho.rst.sctb</td>
<td>79.89</td>
<td>82.74</td>
<td>81.29</td>
</tr>
<tr>
<td>mean</td>
<td><strong>90.74</strong></td>
<td><strong>91.24</strong></td>
<td><strong>90.93</strong></td>
</tr>
</tbody>
</table>

Table 4.4 Final scores for the EDU segmentation task on the treebanked data. The best score has been marked in bold. The model performs the best on the English RST-DT dataset with an F1 score of 97.09% which is the new state-of-the-art score for English.
Table 4.5 Final scores for the EDU segmentation task on the plain tokenized data. The best score has been marked in bold. The model performs the best on the English RST-DT corpus with an F1 score of 95.23%.
Chapter 5

Identification of Discourse Connectives in Hindi

5.1 Introduction

As seen in the previous chapter, we worked on the problem of discourse unit segmentation and connective identification across 11 different languages and presented a multilingual model which returned high F1 scores across all languages. However, Hindi, despite being used and spoken by over 600 million people, was not included in those 11 languages. Noting the lack of a curated dataset for the above mentioned tasks, especially in a multilingual setting, in this chapter, we present a Hindi dataset consisting of 1850 sentences for the identification of explicit discourse connectives.

We obtain the base data from the Hindi Discourse Relation Bank which broadly follows along the lines of the Penn Discourse TreeBank with a slightly modified annotation scheme and three additional grammatical classes for the explicit discourse connectives. We annotate the presence of an explicit discourse connective based on a predefined set of rules which ensures consistency in annotation. We also present a connective identification model borrowing from the transformers based approach mentioned in the previous chapter. We experiment with the multilingual model as well as some Indian language specific models and report our final scores on the new dataset.

5.2 Data Creation

In this section, we will describe our efforts towards creating the curated dataset for the task of identification of explicit discourse connectives in Hindi. The final dataset is created following the CoNLL-U format and consists of 41615 tokens (words) present in 1850 sentences. We will look at the Hindi Discourse Relation Bank from where the base data is sourced. The data was obtained in the SSF format and needed to be converted into the CoNLL format to facilitate the annotation for discourse connectives. We will then discuss the annotation framework, the
tagset and the annotation schema used as well the rules we followed for annotating the presence of an explicit discourse connective.

5.2.1 Sourcing the data from the HDRB

The Hindi Discourse Relations Bank consists of newspaper articles taken from the Hindi newspaper Amar Ujala. Building upon the Penn Discourse TreeBank approach, apart from the three grammatical classes of explicit connectives mentioned in the PDTB (adverbials, subordinating conjunctions, and coordinating conjunctions), the HDRB introduces three new classes for the identification of explicit discourse connectives as follows -

1. **Sentential Relatives**: These refer to relative pronouns that modify verb phrases and connect a relative clause with its matrix clause.

   ![Figure 5.1 Example of sentential relatives](example-image)

   "[He gave his cricket bat to his brother] so that [he could play in the match tomorrow]."

2. **Subordinators**: These refer to suffixes, postpositions and verbal participles that define nonfinite clauses with an abstract object interpretation.

   ![Figure 5.2 Example of subordinators](example-image)

   "Upon [hearing his brother’s words] [Rahul became very sad]."

3. **Particles** These refer to the words which are used to denote the inclusion of adjectives, verbs, adverbs and entities.

   ![Figure 5.3 Example of particles](example-image)

   "[Ram is going out to play today] [Mohan will] also [go with him]."

In the above examples, the words marked in bold represent the discourse connective with its two arguments marked using brackets.

There are several instances in the data where a certain word functions as a discourse connective in one setting but doesn’t in another depending upon the surrounding sentences. Ensuring
that these words, which occur in a non-discourse connective context, are not marked as a discourse connective, is an important task. We shall discuss these scenarios in detail when we describe our annotation schema and rules.

5.2.2 The Shakti Standard Format (SSF)

The Shakti Standard Format is a representation framework which is useful for storing the information obtained from the linguistic analysis of data. It represents the information in a text or sentence in the form of trees along with attribute-value pairs which specify the features/properties of every node in the tree.

The Shakti Standard Format has two main parts - the header and the body. The header stores all the metadata about the file such as the title, date, source, etc. The actual sentences are the part of the body section. The sentences are represented with each line containing one token (word). Each row consists of 4 parts - the index of the word, the word itself, the part of speech tag of the word, and the attribute-value pairs for each word. The attribute-value pairs for each word store the list of morphological features for every token such as the root word, gender, number, person, etc. Figure 5.4 and 5.5 shows an example of the head and body sections of a data snippet from our dataset in SSF.

![Example SSF output](image)

**Figure 5.4** The head section in SSF. Information regarding the source, date, etc. is represented in this block.

Despite being a very useful method of storing linguistic information, in order to annotate the data for discourse connectives and train our connective identifier models on the data, we need to convert the available data from the Shakti Standard Format into the CoNLL-U format which is basically a TSV (tab separated value) representation format of data.
CoNLL-U is a modified version of the CoNLL-X format with each line representing a token (word) with the annotations distributed across 10 tab-separated fields and a blank line marking the sentence boundaries after each sentence. We use the SSF-to-CoNLL conversion tool which is available on the official GitHub repository of LTRC, IIIT-H.

Figure 5.5 The body section in SSF. Each line represents a word in the sentence in 4 parts.

https://github.com/ltrc/SSF-to-CoNLL-Convertor
5.2.3 Data Annotation

Since the data has been sourced from the Hindi Discourse Relation Bank, which follows the lexically grounded approach of the PDTB, the model of coherence relations for this data is the PDTB framework. Therefore, our task in this case corresponds to the identification of the spans of the explicit discourse connectives present in the text.

In order to mark the explicit discourse connectives in the text, we follow the BIO (Beginning, Inside, Outside) tagging format for the tagging the tokens present in the data. Each token is marked with a label that depicts the position of the token with respect to the explicit discourse connective - whether the token is at the beginning of a discourse connective, inside the connective segment or outside the connective segment. Figure 5.6 shows an example of a BIO tagged sentence taken from our dataset.

![Figure 5.6 A sentence annotated using the BIO format](image)

"B" corresponds to the "Seg=B-Conn" tag which marks the beginning of the chunk which contains the discourse connective. "I" corresponds to the "Seg=I-Conn" tag which marks the token in continuation of the connective to its left. Table 5.1 shows the tagset used for annotation along with the description of each tag.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seg=B-Conn</td>
<td>Indicates that the token is at the beginning of an explicit connective</td>
</tr>
<tr>
<td>Seg=I-Conn</td>
<td>Indicates that the token is a continuation of the connective to its left</td>
</tr>
<tr>
<td>_</td>
<td>Indicates that the token is outside the connective segment to its left</td>
</tr>
</tbody>
</table>

Table 5.1 The tagset used to mark the presence of an explicit discourse connective.
In the Hindi language, there are several instances in which a certain word will function as a discourse connective in one occurrence but not in another depending upon the sentences. Figure 5.7 shows an example of this.

**Sentence 1**: राम ने खाना खाया और वह सो गया।
**Translation**: Ram ate food and he went to sleep.

**Sentence 2**: राम और श्याम ने अपना खाना खा लिया।
**Translation**: Ram and Shyam ate their food.

**Figure 5.7** Problem of ambiguity in Hindi discourse connectives

In the above example, the word "और" (and) occurs in both Sentence 1 and Sentence 2. However, it only functions as a discourse connective in Sentence 1 where it joins two phrases. In Sentence 2, it does not function as an explicit discourse connective since it is connecting two NPs.

Handling these instances is an important step during the annotation process to ensure uniform and accurate annotations. We will now describe the set of rules which were followed while annotating the presence of explicit discourse connectives. These rules are derived from the Hindi Discourse Relation Bank Annotation Guidelines which were published by LTRC, IIIT Hyderabad.

As discussed previously, the HDRB adds three more classes of explicit discourse connectives to the three existing classes which were already specified in the PDTB. Therefore, a total of 6 syntactic categories are considered while marking explicit discourse connectives in Hindi. Each category has its own rules for identifying when a given token functions as an explicit discourse connective.

1. **Coordinating Conjunctions** -

These words connect two clauses or phrases which have the same syntactic status. Words like "और" (and), "पर" (but), etc. come under this category. However, as is apparent in Figure 5.7, we do not mark the word as a discourse connective when it connects two NPs. Coordinating conjunctions can also occur in pairs like the words "न केवल ............ बल्कि" (not only ..... but). In these cases, we mark both words of the connective separately. The start of every word is marked as "Seg=B-Conn" with the segment to its left (if any) marked as "Seg=I-Conn". Figure 5.8 will provide further clarity.
2. **Subordinating Conjunctions**

These words connect finite adverbial clauses to their matrix clauses. Usually, these occur at the start of a clause but some might occur in the middle as well. Some examples of subordinating conjunctions include "क्योंकि" (because), "यदि ............. तो" (if ....... then), "इसलिये" (because), etc.

Similar to coordinating conjunctions, as seen in the examples, subordinating conjunctions can also occur in pairs. These instances will be marked in the same way as mentioned above. However, in this case, one of the elements of the pair can be implicit.

3. **Sentential Relatives**

This category consists of relative pronouns which connect a relative clause with its matrix clause. Words like "जिसके कारण" (due to which), "जिससे " (so that), etc. are examples of sentential relatives.

We only mark those instances of sentential relatives that modify verb phrases as discourse connectives. Similar to coordinating conjunctions, we do not mark those instances where noun phrases are being modified.

4. **Adverbials**

Postpositional and adverbial phrases which act as discourse connectives come under this category. "वास्तव में" (actually), "फिर" (then), "वही दूसरी ओर" (on the other hand), "तभी" (then), etc. are some examples of adverbial which function as explicit discourse connectives. Figures 5.9 shows an example of this.
Example 1 -

[Virat Kohli isn't able to score runs. On the other hand, Patidar is scoring a lot of runs.]

Example 2 -

[The power went off at the start of the concert for a bit. Other than this, the event went on smoothly.]

Figure 5.9 Example 1 shows an adverbial phrase whereas Example 2 shows a postpositional phrase as an explicit discourse connective

5. Subordinators -

As mentioned before, these words introduce non-finite subordinate clauses and consist of suffixes, postpositions or particles following verbal participles. We annotate the instances of these words as explicit discourse connectives in our dataset. Examples of subordinators include "-कर", "-के बाद", "-हुए", etc.

Figure 5.2 is an example of a suffix marking a serial verb and acting as a discourse connective. Figure 5.10 shows how postpositions and particles following verbal participles act as explicit discourse connectives.

Figure 5.10 Subordinators functioning as discourse connectives

6. Particles -

There are several types of particles in Hindi. Some examples include "ही", "भी", "ना", "रूँ", etc.

However, we do not annotate every instance of a particle as an explicit discourse connective. The ambiguity issue arises in this case as well. Consider the particle "भी". It functions as an emphatic particle when it emphasizes certain entities in the text. Figure 5.11 shows an example of "भी" acting as an emphatic particle.

After Kohli got out, Dhoni came in to bat, and while playing he kept his calm and played slowly.
Figure 5.11 Emphatic particle

In the above figure, we do not mark the emphatic particle as a discourse connective since it does not connect any elements in the discourse.

Consider Figure 5.3, in this example, the particle connects two abstract objects and thus, we annotate it as an explicit discourse connective.

We present a curated dataset of Hindi data with specific annotations for identifying discourse connectives. We perform our annotations following the above mentioned rules, identify words belonging to the 6 syntactic categories as specified by the HDRB, handle the ambiguous cases where a certain word doesn’t function as a discourse connective and annotate the appropriate tokens as explicit discourse connectives in our dataset. The final dataset statistics are presented in Table 5.2.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Framework</td>
<td>PDTB</td>
</tr>
<tr>
<td>Segmentation Style</td>
<td>Connectives</td>
</tr>
<tr>
<td>Gold Sentence Boundaries</td>
<td>Yes</td>
</tr>
<tr>
<td>Training Tokens</td>
<td>33311</td>
</tr>
<tr>
<td>Training Sentences</td>
<td>1485</td>
</tr>
<tr>
<td>Testing Tokens</td>
<td>8304</td>
</tr>
<tr>
<td>Testing Sentences</td>
<td>365</td>
</tr>
<tr>
<td>Total Tokens</td>
<td>41615</td>
</tr>
<tr>
<td>Total Sentences</td>
<td>1850</td>
</tr>
</tbody>
</table>

Table 5.2 The final dataset statistics for our Hindi dataset created for the task of explicit connective identification in Hindi.

The final dataset is available online on GitHub.

https://github.com/SahilBakshi98/Hindi_Conn_Dataset
5.3 Approach and Methodology

In this section, we describe our models for the task of connective identification on the dataset described in the previous section. We follow the similar approach described in Chapter 4, leveraging the transformer architecture and fine-tuning it for our classification problem. Apart from experimenting with our multilingual BERT model used in the previous chapter, we also test the performance of Indian language specific pre-trained models such as MuRIL and Indic-BERT. Figure 5.12 provides an outline of our approach.

![Outline of our approach](image)

Figure 5.12 Outline of our approach

We follow the same approach towards handling out of vocabulary words as mentioned in Chapter 4 by using the offset alignment to split the input tokens and modifying the corresponding labels accordingly. While working with this dataset, we didn’t have to deal with the issue of handling extremely long input sequences since we have marked sentence boundaries in the dataset. Every input sequence adheres to the 512 token limit that is required for BERT models.
5.3.1 Multilingual Representations for Indian Language (MuRIL)

MuRIL is a multilingual language model which follows the BERT framework and is specifically built for Indian languages [23]. MuRIL has been trained on a total of 17 languages out of which 16 are Indian languages and the 17th is English. The monolingual training data for the 17 languages has been obtained from Wikipedia and the Common Crawl OSCAR corpus. Parallel corpora consisting of translated and transliterated data pairs are also used for training the model.

Similar to BERT, the pre-training of MuRIL involves two language modeling objectives. The first one is the same as the one used for BERT per-training - Masked Language Modeling (MLM) - where the model attempts to predict the masked tokens. Monolingual data is used for this objective. The second language modeling objective used for MuRIL is called Translation Language Modeling (TLM) [28] which is basically an extension of the MLM objective where the parallel corpora of the translated and transliterated data is used. Figure 5.13 provides an illustration of the TLM objective.

![Diagram of Translation Language Modeling](image)

**Figure 5.13** Translation Language Modeling [28].

Similar to MLM, the task in this case is to predict the masked word. But as seen in the figure, where the English sentence and its French translation are shown, the model can refer to both the English and French parts and is encouraged to align the two representations for predicting the masked English word.

5.3.2 IndicBERT

IndicBERT is a multilingual ALBERT model trained on 12 languages which include 11 major Indian languages and English [22]. ALBERT is short for ‘A Lite BERT’. It is a model which tackles the problems of GPU memory limitations and longer training times when the model size becomes too big. ALBERT reduces memory consumption and increases the training speed of the BERT model and has fewer parameters as compared to BERT.
IndicBERT is pretrained on the novel corpus put forth by the model authors themselves known as IndicCorp which consists of over 8 billion tokens and 12 languages. In this case as well, the pretraining task involves the Masked Language Modeling objective which we have seen for both BERT as well as MuRIL. However, the other tasks of Next Sentence Prediction (BERT) or the Translation Language Modeling (MuRIL) are not employed.

As mentioned in the previous chapter, we use the Huggingface transformers library to obtain the models to train on our dataset. Once the token embedding and splitting is complete, along with the label offset alignment, the input data is passed to the model. The output of all the encoders (multilingual BERT, MuRIL, and IndicBERT) is then passed through a classification layer which applies a linear transformation to the input and provides us the discourse connective markers as output.

We use an 80-20 training-testing split for our dataset. We fine-tune the above mentioned pretrained models by presenting the task of discourse connective identification as a token classification problem. We use a batch size of 8 for training, with the input sequences being padded to the length of the longest input sequence of that particular batch. The optimal parameters for our best scores were using an Adam optimizer with a learning rate of 1e-5 and a training period of 15 epochs.

5.4 Results

In this section, we report the precision, recall and F1 scores for all three models on our new dataset. To the best of our knowledge, this is the first instance of discourse connective identification in Hindi in a multilingual setting. The final scores have been reported by averaging the scores obtained over 5 runs.

<table>
<thead>
<tr>
<th>Model</th>
<th>Base Model</th>
<th>Precision</th>
<th>Recall</th>
<th>F1 Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multilingual BERT</td>
<td>BERT</td>
<td>91.2</td>
<td>88.5</td>
<td>89.8</td>
</tr>
<tr>
<td>MuRIL</td>
<td>BERT</td>
<td>89.6</td>
<td>87.8</td>
<td>88.7</td>
</tr>
<tr>
<td>IndicBERT</td>
<td>ALBERT</td>
<td>89.1</td>
<td>89.7</td>
<td>89.4</td>
</tr>
</tbody>
</table>

Table 5.3: The final model scores on our Hindi dataset for the task of explicit connective identification in Hindi.

The multilingual BERT model gives us the best scores with an F1 score of 89.8%, performing narrowly better than the IndicBERT model which gives an F1 score of 89.4%. The final precision, recall and F1 scores have been illustrated in Table 5.3. The subword based tokenization process followed by BERT helps the model build better embeddings and handle the out of vocabulary tokens well which improves the model performance. The model performance of our multilingual model is comparable to the current state of the art discourse connective identifier.
for Hindi whose F1 score is 90.8%. The difference arises in the use of language specific lexical
and dependency features whereas we are more inclined towards presenting a generalized mul-
tilingual approach which helps in making Hindi more available as a language for multilingual
discourse parsing.

In the previous chapter, we have worked with 3 other languages and their datasets which
follow the PDTB formalism on the task of identification of explicit discourse connectives. The
performance comparison of our multilingual BERT model is shown in Table 5.4. Despite having
a significantly less amount of data as compared to those datasets, the model performance for
Hindi is comparable to the scores obtained for English and Turkish and better than the score
on the Chinese dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>No. of tokens</th>
<th>Precision</th>
<th>Recall</th>
<th>F1 Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>English PDTB</td>
<td>1,156,657</td>
<td>90.4</td>
<td>91.9</td>
<td>91.2</td>
</tr>
<tr>
<td>Turkish TDB</td>
<td>496,358</td>
<td>90.4</td>
<td>91.2</td>
<td>90.8</td>
</tr>
<tr>
<td>Chinese CDTB</td>
<td>73,314</td>
<td>85.1</td>
<td>87.5</td>
<td>86.3</td>
</tr>
<tr>
<td>Our Hindi dataset</td>
<td>41,615</td>
<td>91.2</td>
<td>88.5</td>
<td>89.8</td>
</tr>
</tbody>
</table>

Table 5.4 Comparison of the scores obtained by the multilingual BERT model on the newly
created Hindi dataset with the scores on the 3 datasets mentioned in Chapter 4.

5.5 Conclusion

In this chapter, we recognized the lack of a curated dataset in the Hindi language for the
task of explicit connective identification in a multilingual setting. We present a curated dataset
in Hindi to add to the existing pool of datasets that we discussed in Chapter 4. This is a step
forward towards making Hindi more available as a language for multilingual discourse parsing.
We also present a classification model for the aforementioned task and experiment with 2
more Indian language specific models and report the scores obtained on the new dataset. The
multilingual BERT model performs best with an F1 score of 89.8% and since to the best of
our knowledge, this is the first work for Hindi in a multilingual setting, we have established the
baseline for further work and improvements in this subfield.
Chapter 6

Conclusions and Future Work

In this thesis, we looked at the topic of discourse parsing. We described our efforts towards shallow discourse parsing, multilingual discourse unit segmentation and the identification of explicit discourse connectives in Hindi. Discourse unit segmentation and connective identification are the first steps in discourse parsing which further plays a vital role in the development of several NLP tools for tasks such as machine translation, sentiment analysis, text summarization, and so on.

We described the three major frameworks used for modeling coherence relations in text. Based on the framework, we looked at the respective tasks involved in the discourse analysis of the said frameworks. We described the Penn Discourse TreeBank and the three tasks involved in shallow discourse parsing in English. We compared and analysed the approaches towards these tasks and talked about the performance of the current state of the art PDTB based parsers for English. We then touched upon the Hindi Discourse Relation Bank, a project that follows the PDTB approach for discourse parsing in Hindi. We looked at the existing approaches towards connective identification and argument extraction of explicit connectives in Hindi and discussed the current state of the art systems for these tasks in Hindi.

We developed a multilingual model for discourse unit segmentation and connective identification across 16 datasets encompassing 11 languages and the 3 major coherence frameworks as discussed above. We presented a novel approach towards these tasks by approaching the problem as a token classification task and using the BERT pretrained model for classification. We reported a new state of the art score for discourse unit segmentation on the English RST-DT dataset with an F1 score of 97.09%. We also improved the previous best scores on 15 out of the 16 datasets with the model performing well across languages with the lowest F1 score being 81.29%. We also analysed the performance of our model in the presence and absence of gold sentence boundaries.
We created a curated dataset for the task of identification of explicit discourse connectives in Hindi in a multilingual setting. We presented a dataset consisting of 41,615 tokens in 1,850 sentences with the source data being obtained from the Hindi Discourse Relation Bank. We described the annotation framework, tagset, and the annotation rules that we followed while marking the explicit discourse connectives in the dataset. We also presented a model for the identification of these connectives. Apart from the multilingual model described earlier, we also experimented with Indian language specific models like MuRIL and IndicBERT and compared their performance with our initial multilingual model. We also compared the performance of our multilingual model on the newly created Hindi dataset with the 3 existing datasets (English, Turkish, Chinese) used before which follow the same PDTB annotation framework.

Further work can involve analysing the connectives which the model is able to predict correctly and those which the model is missing. This will help in further fine-tuning of the models resulting in better performance. Expanding the current newly created dataset will further help in making Hindi more available a language for multilingual discourse parsing. The universal dependencies format followed while creating this dataset makes it easier to perform further annotations. As the data size increases, this gives the neural systems more information to work with which will help in improving the performance of discourse analysis systems. Since these are first scores obtained on this Hindi dataset in a multilingual setting, we expect the performance to only get better with time.
Related Publications

- A Transformer Based Approach towards Identification of Discourse Unit Segments and Connectives, Sahil Bakshi and Dipti Misra Sharma. EMNLP CODI-DISRPT 2021, pages 13-21, Punta Cana, Dominican Republic, November 2021
Bibliography


