Chapter 2

Related Work

There are two distinct approaches which have been followed for representing syntactic analysis in a treebank. They are: (i) dependency structure (DS) [171, 116] and (ii) phrase structure (PS) [59].

In a PS tree, the leaf nodes are marked with words that are in the sentence, whereas the internal nodes are marked with non-terminal labels. PS takes the successive words and groups them hierarchically into phrases, and then each phrase is given a syntactic label [42]. Below given are examples of phrase structure trees for English and Hindi sentences respectively.

Ex-(2.1) Children eat fruits.

![Phrase Structure Tree for English sentence (Example 2.1)](image-url)
Ex-(2.2) *bachche phal khaate haiM*

child fruit eat.Impf. is
‘Children eat fruits.’

**Figure 2.2** Phrase Structure Tree for Hindi sentence (Example 2.2)

Penn Treebank for English [110], Penn Chinese Treebank [189], Penn Arabic Treebank [107], etc., are some of the well known phrase structure (PS) treebanks.

In a dependency structure (DS) tree, all nodes are marked with words that are in the sentence. In DS, syntactic dependency is a relation between a syntactic head and its arguments and adjuncts. Below are examples of dependency trees for English and Hindi sentences respectively.
Figure 2.3 Dependency Structure Tree for English sentence (Example 2.1)

Figure 2.4 Dependency Structure Tree for Hindi sentence (Example 2.2)

Prague Dependency Tree Bank (PDT) [81], Dependency Treebank for English [145], Turin University Treebank [46], etc., are some of the well known dependent structure (DS) treebanks.

2.1 Phrase Structure Grammar (PSG) Based Treebanks

2.1.1 The Penn English Treebank

Penn Treebank [110] is one of the major and most popular English treebanks. The Penn Treebank has developed nearly 7 million words of part-of-speech tagged text, 3 million words of skeletally parsed text, 2 million words of text parsed for predicate-argument structure, and 1.6 million words of transcribed spoken text annotated for speech disfluencies [170]. It is a human-annotated and partially 'skeletally' parsed corpus consisting of 4.5 million American English words. This treebank has three types of annotation schemes: POS tagging, syntactic bracketing, and disfluency annotation [170]. Its output mostly consists of POS tagged and syntactically bracketed versions of written texts. Earlier,
bracketing was done using simple skeletal parse, after that it used a richer predicate-argument bracketing schema, and finally, it developed a tagged and parsed version of part of the Switchboard corpus of telephone conversations, and also a version annotated for disfluencies [170]. Penn English Treebank [110] has contributed to the outcome of English Part-of-Speech Taggers and Parsers [63, 62, 55]. The success of Penn Treebank led to the creation of treebanks in many languages such as Czech, French, German, Japanese, Polish, Spanish, Arabic, Chinese, Korean, and Turkish [189].

2.1.2 The Penn Chinese Treebank

Following Penn Treebank of English, a **Penn Chinese Treebank** was also created [189]. The aim of the Penn Chinese Treebank project is to create 500,000-word corpus of Chinese text with syntactic bracketing. The Chinese Treebank concentrates on three problems [189]:

(a) The possibility of the word segmentation work, (b) The weak inflectional morphology in Chinese, and (c) Hard constructions in Chinese syntax [189].

2.1.3 The Penn Arabic Treebank

The Penn Arabic Treebank (ATB) has three full parts of morphologically and syntactically annotated data: (1) Part 1: roughly 166K words of written Modern Standard Arabic newswire from the Agence France Presse corpus; (2) Part 2: roughly 144K words from Al-Hayat distributed by Ummah Arabic News Text, and (3) Part 3: roughly 350K words of newswire text from An-Nahar morphologically annotated. The ATB corpora have annotated the following information: morph, part-of-speech, English gloss, and syntactic structure [110, 109, 44]. The goal of Arabic Treebank is:

(i) To be of high quality,

(ii) To have a high level of descriptive consistency, and

(iii) To have credibility with regard to the attitudes and respect for correctness known to be present in the Arab region as well as with respect to the NLP and wider linguistic communities [107].

2.1.4 Building a Treebank for French

French Treebank consists of annotated newspaper corpus of 1 million words [4, 5] with parts of speech, inflections, compounds, lemmas and constituency. The aim was to come up with a theory neutral, surface oriented, and error-free Treebank for French. They have differentiated a tagging and a parsing phase similar to Penn TreeBank [110, 170], and have come up with an automatic annotation process followed by manual validation and correction. Similar to the Prague treebank [24], they have depended on
various types of morphosyntactic and syntactic annotations for which they have created broad guidelines [3].

2.1.5 Developing a Syntactic Annotation Scheme and Tools for a Spanish TreeBank

They have developed specifications and tools inorder to build a Syntactically Annotated Corpus (SAC) of Spanish newspaper texts. The corpus contains 1,500 syntactically annotated sentences taken from newspapers. They developed their own specifications incorporating the appropriate features for Spanish into the general mainstream of corpus annotation. The specifications were based on the understanding acquired in annotating the real text. The first 500 sentences selected were complex and had all the variations that could be found in newspaper texts. Sentences were taken from two different sources for the sake of variety in the phenomena. Three different versions of the specifications were created which allowed additions and improvements, and also allowed to incorporate major modifications to the current guidelines. In the initial stage of the corpus, they developed few tools to aid human annotators with coding and debugging tasks [124].

2.2 Dependency Grammar (DG) Based Treebanks

2.2.1 The Prague Dependency Treebank

Prague Dependency Tree Bank (PDT) for Czech (which has relatively free word order) is a large-scale effort which implements a three-tier annotation scheme and annotates morphological information, analytical, and tectogrammatical level annotations at three levels of analysis [81]. Out of the three levels, the analytical and tectogrammatical level are dependency based. The morphological level uses word forms, tags, and lemmas. The analytical level or surface syntax uses dependencies and analytical functions of dependencies. About 30000 Czech sentences have been annotated manually on this layer. The tectogrammatical level tries to capture the deep-semantics of the sentence. It contains tectogrammatical functions such as Actor, Patient, Addressee, etc.

At the dependency layer, in PDT, there are PP (Prepositional) phrases whose head is a preposition except the genitives. In case of genitives, ‘of’ is not the head of the phrase, instead noun is the head of that phrase. There are no PP (Postpositional) phrases in Hindi Treebank (HyDT). Postpositions occur within the Noun phrases (NPs). In HyDT, the Noun phrase (NP) having a genitive becomes the dependent, i.e., child node of the head NP (parent node) whereas in PDT, head NP is dependent of the genitive NP. In PDT, there is dependency relation between auxiliaries and verbs, prepositions and nouns, and attributive adjectives and nouns whereas in HyDT there is intra-chunk dependency relation between the above pairs.
2.2.2 A Dependency Treebank for English

A Dependency Treebank [145] for English presents the syntactic annotation level of a ISLE project. It aims to richly annotate a small corpus (13,000 words) of dialogs with multiple layers of annotation, i.e., orthography, intonation, syntax, NP coreference, information structure, dialog structure, and discourse structure. It resolves to annotate syntax without any influence of all other layers of annotation, beginning from the speech transcription. In developing the annotation, its objective was to have a simple representation that is beneficial for any syntactic study of this corpus, independently of the application such as parsing or generation [145]. In this treebank, the first line is the word, the second is part-of-speech, the third shows the surface role (SRole). There is dependency relation between auxiliaries and verbs, and determiners and nouns. In control verbs, an empty subject of the embedded verb (PRO in the Chomskyan theories) is added as a new node, whose word is the empty word e. [145]

In Dependency Treebank [145] for English, there is dependency relation between auxiliaries and verbs, and determiners and nouns whereas in Hindi dependency Treebank (HyDT) there is intra-chunk dependency relation between the above pairs.

In Dependency Treebank [145] for English, in control verbs, an empty subject of the embedded verb (PRO in the Chomskyan theories) is added as a new node, whose word is the empty word e. In HyDT, we are inserting NULLS wherever a head node is missing, for example, if there is a missing verb which has its dependents in the sentence, then we insert a NULL for this verb and attach its dependents to this NULL verb. We not only insert null for the missing verbs but also for missing conjuncts, noun, etc., which have their dependents in the sentence.

2.2.3 Turin University Treebank

The purpose of Turin University Treebank (TUT) [46] project was to develop a collection of Italian sentences that have been morphologically and syntactically annotated. It contains the definition of a TUT representation format which shows the peculiarities of Italian and its use to a newspaper corpus by the means of tagging and parsing tools. The TUT representation format is dependency-based paradigm which is centered upon the approach of predicate-argument structure and characterized by a rich grammatical relations system. The development of treebank includes an annotation process performed by a human annotator. The annotator is helped by an interactive parsing tool that builds incrementally syntactic representation of the sentence. A thousand sentences have been automatically annotated through an interactive parsing tool. [46]

2.2.4 The TIGER Treebank

For German language, the first attempt in the direction of treebanks was the NEGRA Corpus ([192] and [191]). The objective of TIGER Treebank [47] is to build the biggest and most exhaustively
annotated treebank for German. It is a corpus of 35,000 syntactically annotated sentences from German newspaper. The annotation format and scheme of TIGER Treebank are based on the NEGRA corpus [47]. But, the TIGER Treebank surpasses the NEGRA corpus in size as well as in detail of annotation [47]. As the NEGRA Corpus is quite limited in its size, the creation of the TIGER Treebank as a comprehensive resource for the German language was an essential attempt to overcome these hindrances. In both NEGRA and TIGER corpus, a hybrid framework is followed which combines the benefits of dependency grammar and phrase structure grammar. This approach is useful for free-word order languages like German, which has a huge variety of discontinuous constituency types [167]. The annotation of the sentences in the treebank is represented on a various number of levels: Word-level: Terminal nodes are labelled with Part-of-speech. Phrase categories are encoded in Non-terminal nodes. Syntactic functions are represented by the edges of a tree [47].

2.2.5 The Alpino Dependency Treebank

The Alpino Dependency Treebank “is a wide-coverage computational analyzer of Dutch” [181]. It intends to parse the unrestricted text accurately and completely [181]. It consists of about 6,000 sentences of news paper text annotated with dependency trees. The purpose behind developing the Alpino Dependency Treebank is the requirement of syntactically annotated corpus to train the grammar and to evaluate its performance. “The Alpino Dependency Treebank is a searchable treebank in an XML format” [181]. The annotation procedure is divided into two parts: First a sentence is parsed with the Alpino parser and then the best parse is selected from the set of generated parses. Various tools that have been developed and implemented in Hdrug, a graphical environment for natural language processing [193], makes easy the two parts of the annotation procedure.

2.2.6 Basque Dependency Treebank

Basque Dependency Treebank [7] is part of a general project\(^1\) which aims to build annotated corpora which is linguistically annotated at syntactic, semantic and pragmatic levels. They syntactically annotate the Eus3LB corpus (25,000 word-forms from EPEC [7] and 25,000 words from newspapers considered as equivalent to the corpora in the other languages in the project) following the dependency-based formalism [8]. For syntactic tagging system, it has adopted the framework given in [195,194]. The tag set explains the most important grammatical structures like relative clauses, causative sentences, coordination, discontinuous elements, elliptic elements, etc. The hierarchy differentiates between various general levels, which are further indicated in subsequent levels. For example, in the general level there are “structurally case-marked complements, modifiers, negation, linking-words, auxiliaries, others and semantic relations” [7].

2.2.7 Russian Dependency Treebank

\(^1\) http://www.dlsi.ua.es/projectes/3lb
The corpus created in the framework of the **Russian Dependency Treebank** [45] project has lot of subcorpora that vary in the level of annotation. The following three levels are:

1) **Lemmatized texts:** For each word, its normal form (lemma) and part of speech are given.

2) **Morphologically tagged texts:** For each word, along with the lemma and the part of speech, a full set of inflectional morphological attributes is mentioned.

3) **Syntactically tagged texts:** Besides the full morphological markup at the word level, each sentence is given a syntactic structure [45].

Russian texts (1,000,000 words) are annotated with dependency structures. The structure contains information about which words of the sentence are syntactically connected, and also relegates each link to one of the several dozen syntactic types [45]. This is an important feature, as the majority of syntactically annotated corpora, those which are already available and those which are under construction represent the syntactic structure through constituents [45].

### 2.2.8 The Quranic Arabic Dependency Treebank

In the **Quranic Arabic Dependency Treebank (QADT)** [66], the Quranic Arabic Annotated Corpus (77,430 words) contains the Arabic grammar, syntax and morphology for each word in the Holy Quran. The corpus has analysis at 3 levels: morphological annotation, a syntactic treebank, and a semantic ontology. Its objective is to build annotated corpora with linguistic annotation at syntactic, semantic and pragmatic levels. The Quranic treebank maps out the entire grammar of the Quran by connecting Arabic words through dependencies.

### 2.2.9 Universal Dependencies

Different languages have different annotation schemes which have obstructed multilingual research on syntax and parsing. This won’t allow carrying out accurate comparative evaluations and cross-lingual learning experiments. Cross-linguistically there has to be consistent annotation as it is very much required for accurate comparative evaluation and cross-lingual learning experiments. This also helps in multilingual system development and comparative linguistic studies. The Universal Dependencies (UD) project solves this issue by creating cross-linguistically consistent treebank annotation for many languages using dependency-based lexicalist framework. UD aims to help: (i) the development of multilingual parsers, (ii) cross-lingual learning, and (iii) parsing research from the viewpoint of language typology. It also aims to pull out the similarities and idiosyncrasies between typologically different languages (morphologically rich, pro-drop, and clitic doubling). The annotation scheme is based on the combination of (universal) Stanford dependencies [198, 197, 196], the universal Google dependency scheme (Universal Dependency Treebanks) [199], the Google universal part-of-speech tags [200], and the Interset interlingua for morphosyntactic tag sets [202] used in the HamleDT
treebanks (modifies the current treebanks using a common annotation scheme, [201]). The broad idea is to present a universal inventory of categories and guidelines to promote consistent annotation of identical constructions across languages and at the same time permitting language-specific extensions when required [203].

There are two layers of annotation in UD which are from different sources. In the morphological layer, the Google universal tag set is used which evolved from the cross-linguistic error analysis based on the CoNLL-X shared task data by McDonald and Nivre [204]. The morphological layer was formed on Interset [202], which was initially used as a tool for conversion between morphosyntactic tag sets of multiple languages. In the syntactic layer, the Stanford dependencies are used which were developed in 2005 which in the course of time came out as the actual standard for dependency analysis of English and from that time they have been adapted to a number of different languages [205, 206, 207, 208, 209] [203].

The Universal Dependency Treebank (UDT) project [210] was the first effort where the Stanford dependencies and the Google universal tags were merged into a universal annotation scheme. It released treebanks for 6 languages: English, French, German, Spanish, Swedish, and Korean in 2013; and 11 languages: Brazilian Portuguese, English, Finnish, French, German, Italian, Indonesian, Japanese, Korean, Spanish and Swedish in 2014; and Tsarfaty [211] was the first one to offer the idea of including morphology [203]. In 20142, the second version of HamleDT [212] presented Stanford/Google annotation for 30 languages and after that there was the development of universal Stanford dependencies (USD) [213]. The new Universal Dependencies is formed by combining all these resources into a single coherent framework, formed on universal Stanford dependencies, an extended version of the Google universal tagset, an updated part of the Interset feature inventory, and an updated version of the CoNLL-X scheme (called CoNLL-U). The annotation guidelines were first released in October 2014 [203].

1) Word Segmentation: There is dependency relation between words in the UD annotation as the annotation is based on a lexicalist view of syntax3. The elementary components of annotation are syntactic words and this indicates that we need to break off and open the contractions. For example, in Spanish the clitics are broken off, i.e., damelo ‘give me it’ is broken into da me lo ‘; whereas in French the contractions are opened, i.e., au = a le ‘. Such instances are considered as multiword tokens as a lone orthographic token corresponds to various words. Since words with spaces are not permitted in UD, the multiword expressions are annotated with the help of certain dependency relations, instead of collapsing various tokens into one [203].

2) Morphology: In UD, there are three levels of representation for the morphological specification of words4:

   a) a lemma: describes the semantic content of the word.
   b) a part-of-speech tag: gives the lexical category of the word.
   c) a set of features: describes the lexical and grammatical features of the word.

There are 17 part-of-speech tags taken from an updated version of the Google universal POS. These tags are classified into open class words, closed class words, and other symbols. These tags are fixed and used for all languages and not all of them are used in all languages. For instance, in all languages, the difference between common nouns (NOUN) and proper nouns (PROP) is not grammaticalized. Each and every morphological feature is linked with a set of

---

2 http://universaldependencies.org/introduction.html
3 http://universaldependencies.org/u/overview/tokenization.html
4 http://universaldependencies.org/u/overview/morphology.html
values, for example, the Number feature takes the values: Singular, Plural, Dual, Collective, etc. Languages choose the part of features and values that are appropriate and they can also include new features and values whenever required [203].

3) Syntax: There are typed dependency relations between words in the syntactic annotation of the UD scheme. The basic dependency structure is a tree form, where a single word is the head of the sentence which is dependent on a notional ROOT and rest of the words are dependent on another word in the sentence. It follows three principles for dependency:

- a) content words are linked by dependency relations;
- b) function words connect to the content word;
- c) punctuation connects to the head of the phrase or clause in which it occurs

UD also permits to add dependencies in an enhanced dependency representation which is a graph structure. The enhanced dependency representation is required to mark external subjects and the external role in relative clauses and to propagate relations over conjunctions [203].

UD has been extended to Indian languages by converting Pānini Dar Dependencies to UD for the Hindi Dependency Treebank (HDTB). This means that the HDTB annotation scheme is converted to UD scheme and in this process the Paninian dependency relations and POS tag set used in Hindi is converted to the UD [214].

The text in the Hindi Treebank is taken from news articles as well as from heritage domain. The Hindi Treebank has: (i)434,856 tokens in 20,783 sentences; (ii)Each sentence has an average of 20.92 words. The treebank is multi-layered and multi-representational [42, 188, 215, 39] and has three layers of annotation: dependency structure (DS) where modified-modifier relations are annotated, PropBankstyle annotation where predicate-argument structure is annotated, and phrase-structure annotation which is independently motivated. Each annotation type has: (i) framework, (ii) annotation scheme, and (iii) annotation guidelines [214].

Dependency Structure annotation is based on the Pānini Grammatical framework [36, 21]. Pānini was a Indian Sanskrit grammarian. The Sanskrit grammar written by Pānini presents a framework for the dependency analysis of a sentence. According to the grammar a sentence is considered as a series of modified-modifier relations and one element, mostly a verb is treated as the primary modified [214].

There are two types of dependency relations existing between words based on the Pānini framework: (i)Kārakas are the direct participants in the action indicated by a verb root. There are six ‘kārakas’, namely ‘kartā’ (the doer), ‘karma’ (the locus of action’s result), ‘karanā’ (instrument), ‘sampradāna’ (recipient), ‘apādāna’ (source), and ‘adhikaraṇa’ (location). (ii)Non-kāraka relations are relations like reason, purpose, possession, etc [214].

There are 3 principles of UD formalism [203] [216]: (i)Content over function: Content words are very important for syntactic representation as the importance is on the dependency relations between content words. This helps to a large extent in getting parallel structures across languages whereas function words in one language frequently match to morphological inflection (or nothing) in other languages. Functional heads rather perform as specifying features of content words, with the help of dedicated relation labels [214]. (ii)Head-first: In spans if it is not at once evident which

---

5 http://universaldependencies.org/u/overview/syntax.html
element is the head then, UD takes the first element in the span as the head, and the other elements attach to the head. This is implemented largely on coordinations, multiword expressions, and proper names. (iii) **Single root attachment:** In every tree, there should be one node having root dependency relation and this should be attached to the artificial root governor [214].

Some of the differences in the design of the two annotation schemes are explained below and also how the current dependency scheme and POS tags of Hindi Treebank map to the Universal dependencies [214]:

a) POS tag-set of UD has 17 tags, whereas the POS tag set of Indian Languages [33], has 32 tags. Almost each of the POS tag of one Treebank has **one to many** mapping with the tags in the other Treebank. The tag WQ from Hindi Treebank which is used to mark question words maps to DET, PRON and ADV in the UD. On the other hand, many POS tags of one Treebank have **many to one** mapping with one tag in the other Treebank. Thus, a lot of tags such as RB, WQ etc. from the Hindi Treebank maps to the POS tag ADV in UD.

b) Multiword names in HDTB are marked by POS tags whereas in UD it is marked at the level of dependency relations.

c) In the reduplicative construction, out of the two words, the first word in the sequence is marked with the tag of its own lexical category and the second word is marked with the RDP tag [33]. There is no such RDP tag in UD for reduplication.

d) All the dependency relations like k3 (instrument of an action), k7t, k7p (time and space respectively of an action), r6 (possession relation between two nouns), map to the label nmod (nominal modifiers) in the UD scheme.

e) In the HDTB scheme, a copular verb is regarded as the head of a copula construction but while converting to UD, predicative nominal in the copula construction is considered as the head and the ‘be’ verb becomes its dependent.

f) In HDTB, either coordinating or subordinating conjunction is considered as the head of the clause. On the other hand, in UD, the first element of the coordinated construction is marked as the head.

There are three components in the conversion process. The first module deals with ellipsis, copula constructions, multiword names and conjunctions and harmonizes the differences of structure from HDTB to UD. The other modules, i.e., second and third module converts POS and Dependency relations of HDTB to UD, respectively. The conversion is done depending on specific heuristics which have conditions related to lexical, structural, morphological information and POS tags [214].
2.2.10 HamleDT—a Harmonized Multi-Language Dependency Treebank

Zeman et al. [217] present HamleDT—a Harmonized Multi-Language Dependency Treebank which is a collection of existing dependency treebanks. All these treebanks are converted in order to accommodate to one type of annotation style. This paper [217] presents a detailed study of many features that can be compared across languages, even if each language has its own style of annotation. Zeman et al. [217] believe that conversion methods can be created which automatically identify such features and then transform them to a single annotation style. This becomes advantageous for comparative corpus linguistics and for machine learning of syntactic parsing [217].

2.2.11 The Persian Universal Dependency Treebank (Persian UD)

The Persian Universal Dependency Treebank (Persian UD) [203] is the transformed version of the Uppsala Persian Dependency Treebank (UPDT) [218]. The initial annotation scheme of this treebank is based on Stanford Typed Dependencies [198, 197]. This annotation scheme was later extended to be used for Persian so that the language specific syntactic relations could be incorporated which was not included in the basic annotation scheme that was developed for English. The treebank contains 6000 sentences which are written texts from different domains, i.e., with different genres (newspaper articles, fictions, technical descriptions, etc.) and tokenization. The differences in the tokenization are because of the orthographic differences of compound words and fixed expressions in the language [203].

Besides the universal annotation scheme and the general rules in the UD, the Persian UD and the UPDT vary in tokenization too. In the UPDT, words having unsegmented clitics are annotated using complex labels whereas in the Persian UD, these words are separated from the clitics and are marked with different labels [203].

The conversion of the UPDT to the Universal Dependencies is done semi-automatically. A conversion script is used in order to reverse the head and dependent relations in the prepositional modifier (prep) and object of a preposition (pobj). Persian specific scripts were used to separate various types of clitics from their host. After that different rules were added for rewriting the coarse-grained part-of-speech tags and the dependency labels. The current release comprises of just a subset of the morphological features [203].

2.2.12 The Ancient Greek and Latin Dependency Treebanks

Bamman et al. [219] explains the evolution, design, and various usages of the Ancient Greek and Latin Dependency Treebanks, enormous compilation of Classical texts where each word has been marked for the syntactic, morphological and lexical information. More than 350,000 words were annotated from the works of Homer’s Iliad and Odyssey, Sophocles’ Ajax, etc. Such an annotated corpus is valued for Classical philology as it allows for morphosyntactic searching and also permits enormous amount of downstream tasks, such as inducing the syntactic behavior of lexemes and automatically recognizing identical passages in between texts [219].
2.2.13 A Gold Standard Dependency Corpus for English

Using the Stanford Dependencies standard, Silveira et al [220] has developed a gold standard annotation of syntactic dependencies in the English Web Treebank corpus. This resource becomes helpful as it becomes a gold standard dependency treebank for English which was lacking earlier and also solves the problem of inadequate availability of gold standard syntactic annotations for informal genres of English text. Silveira et al [220] also present experiments on the application of this resource such as for training dependency parsers and for assessing dependency parsers similar to that which is included as part of the Stanford Parser. Silveira et al [220] show that training a dependency parser on a mix of newswire and web data boosts the performance on that particular type of data and doesn’t highly affect the performance on newswire text, as a result the gold standard annotations for non-canonical text can become important for parsing in general. Moreover, the systematic annotation work has provided information to the SD formalism and its implementation in the Stanford Parser’s dependency converter. The Stanford Dependencies standard was modified and extended, and also the Stanford Parser’s dependency converter was enhanced following the difficulties faced by annotators in the EWT corpus [220].

2.2.14 Multi-view Chinese Treebanking

Qui et al [221] introduce a Chinese treebanking with multi-view annotation framework. It uses dependency structures as the base view and allows conversion into phrase structures with very less loss of information. This treebank was created under the recommended framework and there were 14,463 sentences in the first release (PMT 1.0) which was freely available. The efficiency of the multi-view framework was validated by carrying out an arc-standard transition-based dependency parser and by adding phrase structure features generated by the phrase structure view. Through experimental outcome, it was proved that the extra features for dependency parsing were very effective. More experiments were carried out on dependency-to-string machine translation and proved that this treebank and parser could attain same results in comparison to the Stanford Parser trained on CTB 7.0 [221].

The other dependency treebanks are: Prague Arabic Dependency Treebank (PADT) [168], Danish Dependency Treebank (DDT) [103], Turkish Dependency Treebank [138], Greek Dependency Treebank (GDT) [143], etc.

2.2.15 Dependency Treebanks in Indian Languages

While the work on the current thesis was going on, treebank task for some other Indian languages was also initiated. Thus, treebanks have been developed for Indian languages, such as Telugu, Bangla, and Tamil. For developing syntactic treebanks for Indian languages, dependency-based annotation scheme [21] was used. The treebanks for Telugu, Bangla, and Tamil were created after construction of Hindi treebank (HyDT – Hyderabad Dependency Treebank) [21]. Also, currently, another treebank [39,
42, 188] for Hindi is finished. The Hindi treebank is multilayered where the first layer is a dependency structure (DS) layer. DS annotation in this treebank is also Computational Paninian Grammar (CPG) based.

### 2.2.15 Telugu Treebank

In **Telugu treebank** [182] 1487 sentences were manually pos-tagged and chunked [33], and then dependency relations were annotated. *karaka* relations (participants in an action given in Paninian framework) were marked between chunk heads, for showing the modifier-modified relationship. In this work, certain linguistic constructions have been discussed in detail:

1) **Genitives:** In Telugu the genitive marker is often dropped.

2) **Conjuncts:** Different constructions where a conjunct presence is explicit/implicit.

3) **Copula:** Missing verbs i.e., verbs are dropped.

4) **“ani” (quotative marker in Telugu) constructions**: Various ways of using the lexical item “ani” in language.

Telugu treebank effort also shows how some of the annotation decisions have affected the development of a Telugu parser.

### 2.2.15.2 Tamil dependency Treebank

**Tamil Dependency Treebank version 0.1 (TamilTB.v0.1)** is an effort towards creating a syntactically annotated corpora for Tamil language. It has 600 sentences which are manually annotated for morphology and dependency syntax following the Prague Dependency Treebank. The important goals of **Tamil Dependency Treebank v0.1 (TamilTB.v0.1)** [222] contain: (i) word and syntactic level annotation, (ii) Trying to have highest possible linguistic description at each annotation level, and (iii) Coming up with a huge corpus which is automatically annotated.

The text is preprocessed first and then it is annotated. Preprocessing of the data involves transliteration, sentence segmentation, and tokenization. The annotation scheme of TamilTB.v0.1 is same as that of **Prague Dependency Treebank 2.0 (PDT 2.0)** [80]. The annotation in PDT 2.0 is done on 3 layers: (i) morphological layer (m-layer), (ii) analytical layer (a-layer) and (iii) tectogrammatical layer (t-layer). Right now, TamilTB.v0.1 contains annotation on two layers, i.e., morphological layer (m-layer) and analytical layer (a-layer). Morphological layer (m-layer) is used to mark POS tags or morphological tags to each word in the sentence. Analytical layer (a-layer) aims to mark syntactic information of a sentence. Analytical layer (a-layer) annotation has two stages where: (i) dependency structure of the sentence is captured in a tree form and (ii) the relationship between words are established in the tree [222.2.15.3 **Bengali Treebank**

14
This paper [223] presents a dependency annotation scheme for Bangla treebank and discusses the analysis done on the dependency relations used for Bangla language. They have classified the dependency relations in three different stages, i.e., intrachunk relations, interchunk relations and interclause relations. The above three relations have been again sub-classified and based on this, an annotation scheme was developed. While defining the relations they have dealt with both syntactic and semantic features. They have annotated a corpus containing 4167 Bangla sentences through which they have justified their annotation scheme for creating the treebank (KGPBenTreebank). There are 63 syntactico–semantic relations in the Bangla annotation scheme [223].

2.2.15.4 Towards building a Kashmiri Treebank: Setting up the Annotation Pipeline

There are very few computational and language resources available for carrying out NLP tasks for Kashmiri language. Bhat et al [224] have worked on the basic version of the Kashmiri Dependency Treebank. It consists of 1,000 sentences (17,462 tokens) which have part-of-speech (POS) annotation, and chunk and dependency information. Paninian Computational Grammar (PCG) formalism [21, 28] has been used to annotate the sentences manually in the Treebank. This treebank [224] is an extended version of the pilot Kashmiri Treebank comprising of small corpora of 500 sentences [225], created with the intention of developing guidelines for annotation. Bhat et al [224] present revised version of guidelines with considerable modifications and have done inter-annotator agreement analysis to verify its quality. Bhat et al [224] also talks about a dependency parsing pipeline which has a tokenizer, a stemmer, a POS tagger, a chunker and an inter-chunk dependency parser [224].

In the next chapter, we discuss about the Grammatical model, i.e., Paninian Grammatical Framework which forms the basis for the dependency annotation scheme used for annotating the Hindi Dependency Treebank (HyDT - Hyderabad Dependency Treebank) [21].