INTERLINGUAL SYNTACTIC PARSING: AN OPTIMIZED HEAD-DRIVEN PARSING FOR ENGLISH TO INDIAN LANGUAGE MACHINE TRANSLATION

Pavan Kurariya, Prashant Chaudhary, Jahnavi Bodhankar, Lenali Singh and Ajai Kumar

Centre for Development of Advanced Computing, Pune, India

ABSTRACT

In the era of Artificial Intelligence (AI), significant progress has been made by enabling machines to understand and communicate in human languages. Central to this progress are parsers, which play a vital role in syntactic analysis and support various Natural language Processing (NLP) applications, including Machine Translation and sentiment analysis. This paper introduces a robust implementation of an optimized Head-Driven Parser designed to advance NLP capabilities beyond the limitations of traditional Lexicalized Tree Adjoining Grammar (L-TAG) based Parser. Traditional parser, while effective, often struggle with the capturing complexities of natural languages, especially translation between English to Indian languages. By leveraging Bi-directional approach and Head-Driven techniques, this research offers a revolutionary enhancement in parsing frameworks. This method not only improves performance in syntactic analysis but also facilitates complex tasks such as discourse analysis and semantic parsing. This research involves experimentation the Bi-Directional Parser on a dataset of 15,000 sentences, resulting a reduction in derivation variations compared to conventional TAG Parsers. This advancement highlights how Head-Driven Parsing can overcome traditional constraints and provide more reliable linguistic analysis. The paper demonstrates how this new implementation not only builds on the strengths of L-TAG but also addresses its limitations and contributes to expanding the scope of Tree Adjoining Grammar-based methodologies and advancing the field of Machine Translation.

KEYWORDS

Artificial intelligence (AI), Natural Language Processing (NLP), Tree Adjoining Grammar (TAG), L-TAG (Lexicalized Tree Adjoining Grammar)

1. INTRODUCTION

The rapid progress of machine translation (MT) technology has transformed human communication by permitting a seamless flow of information across linguistic boundaries. Classical machine translation (MT) systems generate translations using rule-based methods, Statistical Models, or neural networks, The intricacies of human languages are still difficult for these methods to fully capture nuances of Indian languages, especially for Low Resource Languages. Head-Driven parsing can be emerged as a significant Parsing Technique that can transform traditional Parser by utilizing the Bi-Directional method to perform computations at levels that were previously unattainable. This research introduces a Head-Driven Bi-Directional parsing for language translation to explore the potential advantages of bottom-up traversal. A traditional parser works from the left and typically requires three inputs: an unknown end position, a given start position, and a Part-of-Speech that has to be parsed. Two pairs of positions are provided by the algorithm in a bidirectional parser: one pair of indices shows the extreme positions between which the category must be identified, and the other pair of indices provides

DIO:10.5121/ijnlc.2024.13402

the precise position of the category once it has been identified. One of the extreme positions corresponds to the actual situation, depending on whether we are parsing to the left or the right.

Parsing is initiated by making top-down predictions on certain nodes and proceeds by moving bottom-up from the head-corner associated with the goal node (root node). In parsing right siblings are parsed from left to right and left siblings are parsed from right to left. Our objective is capturing the nuances of syntactic structure by integrating Bi-Directional Tree traversal into the existing machine translation architecture. The purpose of this study is to investigate the potential benefits of Head-Driven- approaches in improving translation accuracy, fluency, and efficiency.

2. LITERATURE SURVEY

In the early years of Machine Translation, parsing large-scale grammars posed a significant challenge to researchers in the field of Natural Language Processing (NLP). Joshi's imperial work on Tree Adjoining Grammar (TAG) [1] emerged as a pioneering solution, offering a framework that facilitated the parsing of complex linguistic structures. Building upon Joshi's foundation, early endeavours in NLP research also saw the implementation of the Early type Parser, originally proposed by Vijay-Shanker [2], which further enriched the TAG Parser available to computational linguists. Furthermore, in pursuit of language-agnostic solutions, we were inspired to develop a Language-Independent Generator [3] for Natural Languages, aiming to transcend linguistic boundaries and enhance the versatility of computational models. This endeavor broadened the applicability of NLP techniques and contributed to the optimization of Tree Adjoining Grammar-based Machine Translation systems [4][5], fostering advancements in cross-linguistic communication. Continuing the trajectory of innovation within the TAG framework, the research community delved into exploring the full potential of TAG structures. This quest led us to conceptualize vTAG [6], an initiative focused on discovering fresh insights and capabilities inherent in TAG formalisms. Additionally, we introduced sTAG [7] enriching the discourse on TAG-based parsing and generation techniques. A substantial amount of work has been done with a variety of parsing approaches, laying the foundation for real-world applications. Early rule-based approaches, most notably Chomsky's transformational grammar, provided foundational principles for syntactic analysis [8]. Beyond rule-based approaches, Conditional Random Fields (CRFs) and Hidden Markov Models (HMMs), revolutionized parsing by enabling parsers to learn from given corpora [9]. Dependency parsing also emerged as an effective alternative to traditional parsing, offering simpler yet effective representations of syntactic structures [10]. In recent years, Head-Driven parsing has gained attention for its emphasis on hierarchical structures and the identification of key syntactic heads [11]. The integration of linguistic principles, such as Tree Adjoining Grammar (TAG), with machine learning techniques has shown promise in addressing the limitations of traditional parsers, particularly in cross-linguistic parsing scenarios [12]. Bi-Directional parsing methods, as proposed in [13][14], represent a paradigm shift in NLP, offering enhanced capabilities to capture a broader range of syntactic phenomena through both left-to-right and right-to-left parsing strategies. These advancements in parsing techniques have profound implications for various NLP applications, including machine translation, corpus analysis and classification, and information retrieval [15]. Through various experimentation and evaluations, researchers continue to push the boundaries of computational linguistics, shaping the future of NLP and advancing our understanding of human language.

3. IMPLEMENTATION OF BI-DIRECTIONAL HEAD-DRIVEN PARSING FOR TRANSLATING ENGLISH TO INDIAN LANGUAGES

Tree Adjoining Grammar (TAG) is a highly expressive formalism used in computational linguistics for syntactic analysis of Natural Languages. Combining TAG with Bi-Directional Head-Driven parsing creates a powerful method for translating English to Indian languages. Figure 1 depicts the comprehensive pipeline of English to Indian Languages Machine Translation, accompanied by concise descriptions outlining the fundamental NLP components of the translation system.

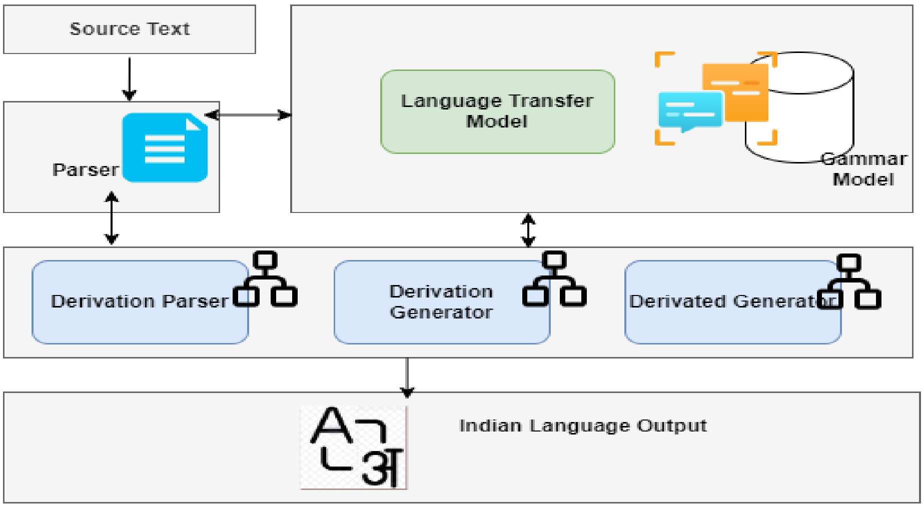


Figure. 1: Bidirectional Head-Driven Parsing-based Machine Translation System

3.1. Pre-Processing

Pre-processing of source sentences in machine translation involves several critical steps to ensure accurate and contextually appropriate translations. The process begins with exploding hyphens and commas, which splits compound words connected by hyphens into individual words and separates items in lists connected by commas. Next, the Date Patterns Identifier detects and normalizes date formats into a consistent structure, facilitating the correct translation of date-related information. The Number Marker then identifies and tags numerical values, ensuring they are preserved accurately in the translation. Noun Marker follows by tagging nouns to help maintain their meaning and context. Phrase Marker is used to identify and mark idiomatic expressions or multi-word phrases that need to be treated as single units to retain their specific meanings. Finally, Transliteration converts words from the source script to the target script, preserving phonetic properties for proper nouns, brand names, or words without direct translations. Together, these pre-processing steps enhance the machine translation system's ability to handle complex linguistic elements, ensuring a more precise and coherent translation.

3.2. Pre-Parser Module

The pre-parser module in natural language processing plays a pivotal role in preparing text for further linguistic analysis and understanding. It includes three essential components: the Part-of-Speech (POS) Tagger, POS Relocation, and the Chunker. Every word in a phrase, including nouns, verbs, adjectives, and so on, must have their parts of speech assigned by the POS Tagger in order to provide an understanding of the grammatical structure. Following this, the POS Relocation step adjusts the positioning of these tags to resolve ambiguities and correct inaccuracies, ensuring that the grammatical roles assigned during POS tagging align correctly with the context. Last but not least, the Chunker converts word sequences into meaningful units that correspond to the grammatical structure of the sentence, such as noun or verb phrases. This chunking process is crucial for understanding the hierarchical relationships within the text and facilitating more advanced parsing tasks. Together, these components of the pre-parser module enhance the system's ability to interpret and process natural language accurately, laying a strong foundation for effective linguistic analysis and subsequent natural language processing tasks.

3.3. Translation Engine

3.3.1. Bidirectional Head-Driven Parser

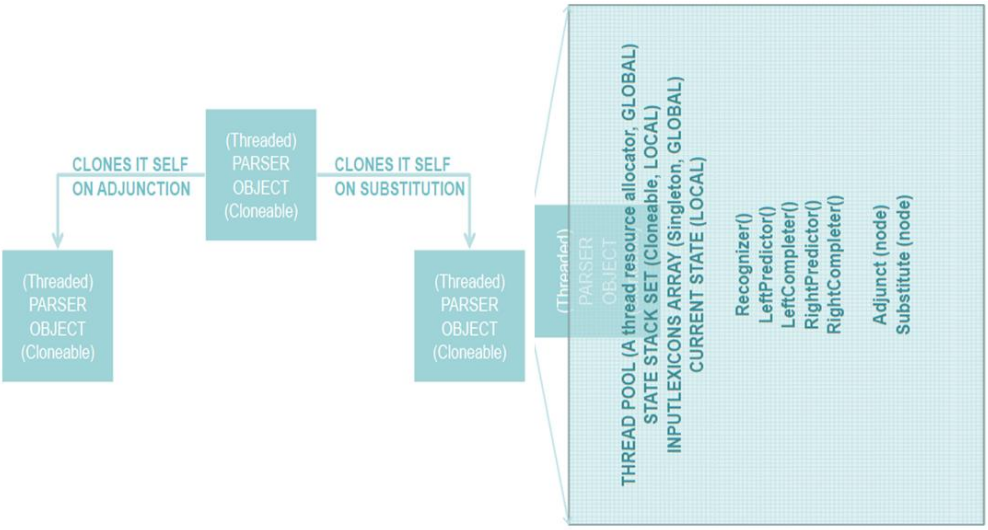


Figure 2: Multithreaded Bidirectional Head-Driven Parser

In Bidirectional Head-Driven Parsing, tree vector serves as a crucial structure for both parsing and generating outputs. Think of it as a reservoir of trees specifically designed for Tree Adjoining Grammars (TAG), where lexicalized trees are drawn for parsing and generation processes. This structure, known as the tree vector, is implicitly defined and essential for the parser's operations. It manages mappings between trees, their names, and lexicons and incorporates a string array that stores the segmented sentence, with each word acting as a key in the mapping.

Figure 1 depicts The Multithreaded Bidirectional Head-Driven Parser, designed for constraint-based Lexicalized Tree-Adjoining Grammars (L-TAG) with multithreading capabilities. Parser selects a node in an elementary tree—using a lexical node for an initial tree and a foot node for an auxiliary tree—and treats it as the <Head>. Parsing begins with top-down predictions on specific nodes and proceeds bottom-up from the Head Node associated with the goal node (root node). During parsing, right siblings are parsed from left to right, while left siblings are parsed from right to left. The use of multithreading enhances the parser's efficiency and speed by allowing multiple parsing operations to be conducted simultaneously. Figure 2 demonstrates the Bi-Directional Head-Driven Parsing process incorporating substitution, adjunction operations, and the generation of a state chart.

Implementation of <head> driven TAG Parser utilizes the close boundary information at various depths of the Natural Languages. This Parser works on TAG Derivation which is an expended tree form of Source Sentence. A hierarchal paradigm of source structure defines the interrelation of its children nodes. This approach takes the benefit of inter-dependency of siblings under a parent/Head, So the generation rules apply at depth from n, n-1 …. Till 0.

Finally, reaching at depth 0 of the TAG parsed derived tree, re-frames the structure into Target structure as depicted in Figure 3. One typical way of defining head grammars is to replace the terminal strings of CFGs with indexed terminal strings, where the index denotes the "head" word of the string. Thus, for example, a CF rule such as $A \longrightarrow abc$ might instead be $A \longrightarrow (abc, 0)$, where the 0th terminal, the a, is the head of the resulting terminal string. For convenience of notation, such a rule could be written as just the terminal string, with the head terminal denoted by some sort of mark, as in $A \longrightarrow \hat{a}bc$.

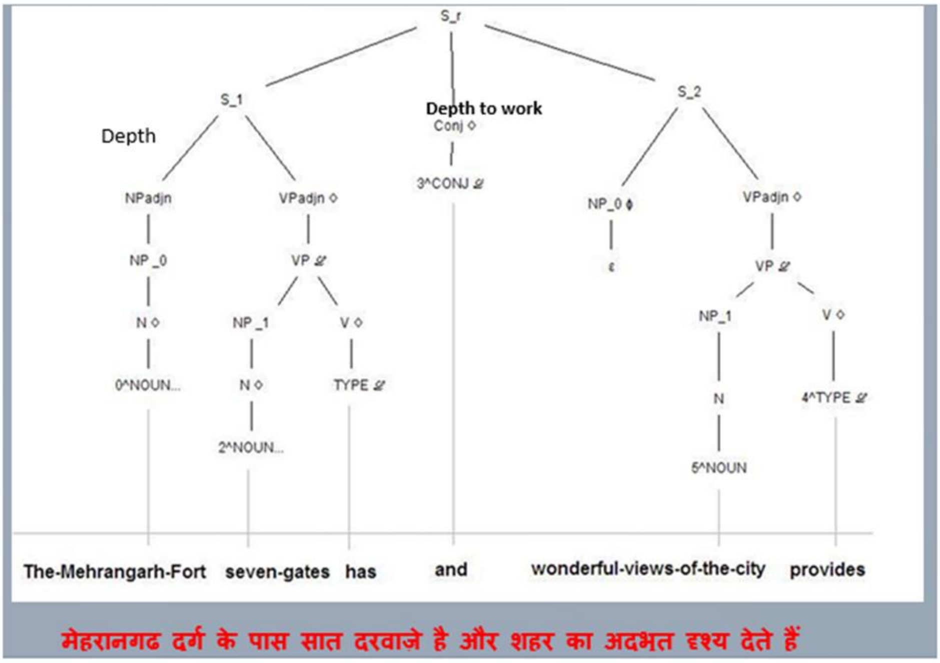


Figure 3: Derived Trees produced by Head-Driven Parser

Two fundamental operations are then added to all rewrite rules: wrapping and concatenation. Operations on Headed Strings

Wrapping is an operation on two headed strings defined as follows:

Let $\alpha\hat{x}\beta$ and $\gamma\hat{y}\delta$ be terminal strings headed by x and y, respectively.

$$w(\alpha\hat{x}\beta, \gamma\hat{y}\delta) = \alpha x\gamma\hat{y}\delta\beta$$

Concatenation is a family of operations on n 0 headed strings, defined for n = 1, 2, 3 as follows:

Let $\alpha\hat{x}\beta$, $\gamma\hat{y}\delta$, and $\zeta\hat{z}\eta$ be terminal strings headed by x, y, and z, respectively.

$$c_{1,0}(\alpha\hat{x}\beta) = \alpha\hat{x}\beta$$
$$c_{2,0}(\alpha\hat{x}\beta, \gamma\hat{y}\delta) = \alpha\hat{x}\beta\gamma y\delta$$

And so on for $c_{m,n} : 1 \leq n < m$. One can sum up the pattern here simply as "concatenate some number of terminal strings m, with the head of string n designated as the head of the resulting string".

It has two properties of composition functions, linearity and regularity. A function defined as f(x1,...,xn) = ... is linear if and only if each variable appears at most once on either side of the =, making f(x) = g(x,y) linear but not f(x) = g(x,x). A function defined as f(x1,...,xn) = ... is regular if the left hand side and right hand side have exactly the same variables, making f(x,y) = g(y,x) regular but not f(x) = g(x,y) or f(x,y) = g(x).

4. EXPERIMENT WITH A BI-DIRECTIONAL HEAD-DRIVEN PARSER WITH TREE BANK

In this section, illustrates the experimentation of the Bi-Directional head-driven parser-based translation system with the source sentence (English) which passes through pre-processing, POS tagging, Parsing, and translating into a target sentence (Hindi). In order to analyze the effectiveness of the Bi-Directional Head-Driven Parser, which makes use of a Multilingual Grammar developed by language specialists, a specialized experimental setup has been established, as illustrated in Fig. 4. Throughout these experiments, we closely monitored CPU usage and memory utilization. A dataset consisting of 15,000 sentences was employed for this purpose. Notably, we compared the performance of the Bi-Directional Head-Driven Parser with that of our previously implemented 'Early TAG Parser', particularly focusing on longer sentences, as illustrated in Fig. 5. The following are the outcomes of these experiments.

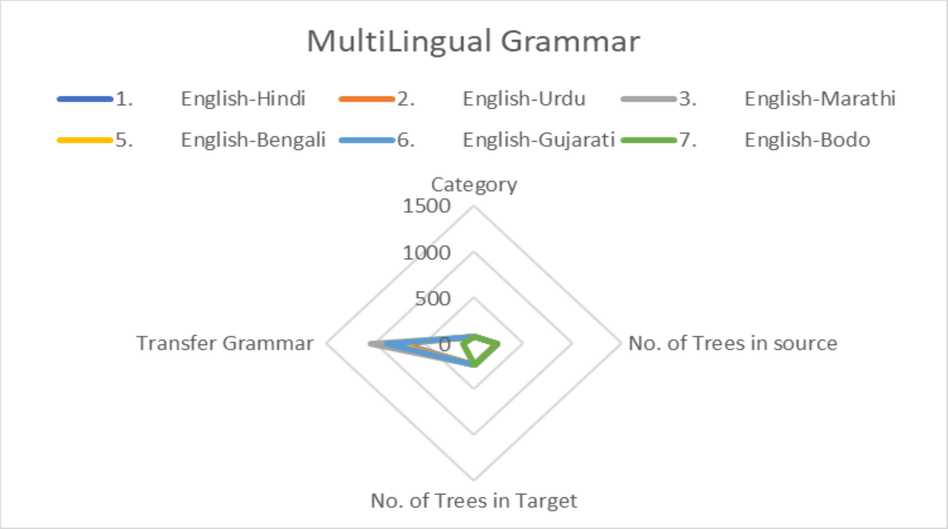
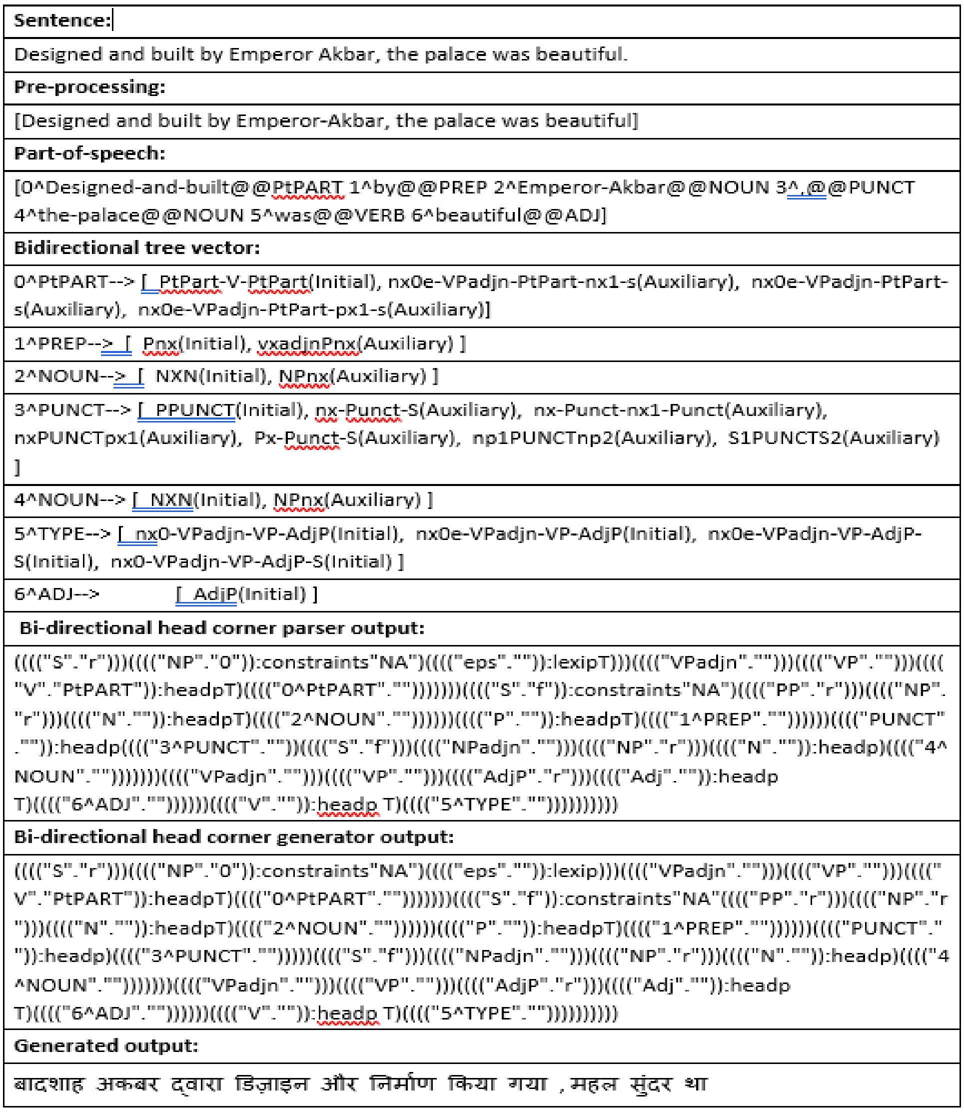


Figure 4: Multi-Lingual Tree Bank for Bi-Directional Head-Driven Parser



Sentence:

Designed and built by Emperor Akbar, the palace was beautiful.

Pre-processing:

[Designed and built by Emperor-Akbar, the palace was beautiful]

Part-of-speech:

[0^Designed-and-built@@PtPART 1^by@@PREP 2^Emperor-Akbar@@NOUN 3^,@@PUNCT 4^the-palace@@NOUN 5^was@@VERB 6^beautiful@@ADJ]

Bidirectional tree vector:

0^PtPART--> [PtPart-V-PtPart(Initial), nx0e-VPadjn-PtPart-nx1-s(Auxiliary), nx0e-VPadjn-PtPart-s(Auxiliary), nx0e-VPadjn-PtPart-px1-s(Auxiliary)]

1^PREP--> [Pnx(Initial), vxadjnPnx(Auxiliary)]

2^NOUN--> [NXN(Initial), NPnx(Auxiliary)]

3^PUNCT--> [PPUNCT(Initial), nx-Punct-S(Auxiliary), nx-Punct-nx1-Punct(Auxiliary), nxPUNCTpx1(Auxiliary), Px-Punct-S(Auxiliary), np1PUNCTnp2(Auxiliary), S1PUNCTS2(Auxiliary)]

4^NOUN--> [NXN(Initial), NPnx(Auxiliary)]

5^TYPE--> [nx0-VPadjn-VP-AdjP(Initial), nx0e-VPadjn-VP-AdjP(Initial), nx0e-VPadjn-VP-AdjP-S(Initial), nx0-VPadjn-VP-AdjP-S(Initial)]

6^ADJ--> [AdjP(Initial)]

Bi-directional head corner parser output:

```
(((("S"."r")))(((("NP"."0")):constraints"NA")(((("eps"."")):lexipT)))(((("VPadjn".""))(((("VP".""))((((
"V"."PtPART")):headpT)(((("0^PtPART".""))))))(((("S"."f")):constraints"NA")(((("PP"."r")))(((("NP".
"r")))(((("N"."")):headpT)(((("2^NOUN"."")))))(((("P"."")):headpT)(((("1^PREP"."")))))(((("PUNCT"
."")):headp(((("3^PUNCT"."")))(((("S"."f")))(((("NPadjn"."")))(((("NP"."r")))(((("N"."")):headp)(((("4^
NOUN"."")))))))(((("VPadjn"."")))(((("VP"."")))(((("AdjP"."r")))(((("Adj"."")):headp
T)(((("6^ADJ"."")))))(((("V"."")):headp T)(((("5^TYPE"."")))))))))
```

Bi-directional head corner generator output:

```
(((("S"."r")))(((("NP"."0")):constraints"NA")(((("eps"."")):lexip)))(((("VPadjn"."")))(((("VP"."")))(((("
V"."PtPART")):headpT)(((("0^PtPART"."")))))))(((("S"."f")):constraints"NA"(((("PP"."r")))(((("NP"."r
")))(((("N"."")):headpT)(((("2^NOUN"."")))))(((("P"."")):headpT)(((("1^PREP"."")))))(((("PUNCT"."
")):headp)(((("3^PUNCT"."")))))(((("S"."f")))(((("NPadjn"."")))(((("NP"."r")))(((("N"."")):headp)(((("4
^NOUN"."")))))))(((("VPadjn"."")))(((("VP"."")))(((("AdjP"."r")))(((("Adj"."")):headp
T)(((("6^ADJ"."")))))(((("V"."")):headp T)(((("5^TYPE"."")))))))))
```

Generated output:

बादशाह अकबर द्वारा डिज़ाइन और निर्माण किया गया , महल सुंदर था

Figure 5: Machine Translation Process using Bidirectional Head-Driven Parser

- 11500 out of 15000 sentences have been successfully Parse and generated on given grammar
- 4531 out of 11500 Sentences having multiple parse derivations.
- 3000 sentences having better output in comparison to the existing Multi-Threaded TAG Parser
- The performance of the Parser has been examined, and it was observed that it requires approximately 40 minutes to parse a total of 11,500 sentences. In comparison, the existing "Early Type Parser" takes around 120 minutes to parse an equivalent set of sentences

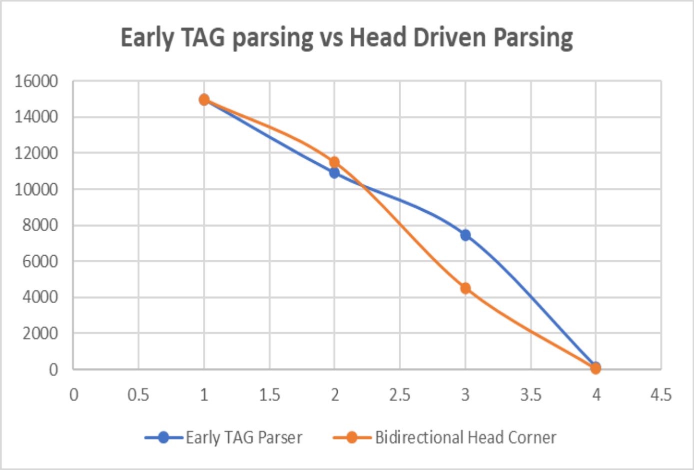


Figure 6: Comparison between ‘Early TAG’ Parser vs. head-driven Parser Performance

5. FUTURE SCOPE

We have also explored the extension of Bi-directional Head-Driven parser by leveraging the unique computational properties of quantum systems to enhance parsing efficiency and accuracy. One potential avenue for enhancement lies in utilizing quantum parallelism, wherein quantum bits (qubits) can represent multiple states simultaneously. By encoding parsing rules and states into qubits, the parser can explore multiple parsing paths concurrently, leading to exponential speedup compared to classical computation. Furthermore, quantum entanglement, which enables correlations between qubits regardless of distance, can facilitate more robust parsing algorithm by capturing long-distance dependencies between linguistic elements. This feature allows for more nuanced and context-aware parsing decisions, leading to improved accuracy, especially in scenarios involving ambiguous or context-sensitive grammatical structures. Additionally, quantum annealing can be employed to fine-tune parsing parameters and optimize parsing strategies can help overcome computational bottlenecks and improve the overall performance of the Head Driven parser. Combined, these quantum-enhanced techniques hold the promise of revolutionizing natural language processing tasks by enabling more efficient and accurate parsing of complex linguistic structures.

The Quantum inspired Head-Driven Parser is able to analyze several linguistic rules at once by taking advantage of the inherent parallelism and uncertainty of quantum computing, in contrast to conventional parsing algorithms that depend on deterministic rules and sequential processing. Through the use of parallel exploration, the parser may evaluate numerous syntactic structures simultaneously, resulting in a significant reduction in computing overhead and the possibility of parsing long and complex paragraphs with exceptional efficiency. Furthermore, by using a quantum-inspired methodology, the TAG Parser is able to identify context and inter connected information in natural language that would be missed by traditional parsing methods.

6. CONCLUSIONS

In this paper, we have analyzed the limitations of conventional TAG Parser and explored the advancements in parsing techniques introduced by recent research. Our research presents the implementation of the Head-Driven (Bi-Directional) Parser, detailing its advantages over traditional TAG parsing methodologies. Through extensive experimentation, we applied this Parser to a multilingual Tree Grammar and conducted empirical tests with 15,000 English sentences from the General domain. The results demonstrate that the Bi-Directional Head-Driven Parser achieved a notable reduction in the variety of parse derivations and exhibited superior performance with multi-clause structures compared to the conventional TAG Parser. Bi-Directional Head-Driven Parser results underscores the effectiveness of the Parser in improving syntactic accuracy and efficiency, particularly in complex sentence structures. Furthermore, our exploration into end-to-end Machine Translation Systems using this Parser highlights its potential for enhancing language processing capabilities.

Despite the rise of Large Language Models (LLMs) in natural language processing, which offer substantial improvements in various tasks, this research reinforces the importance of Bi-Directional Head-Driven parsing approaches. LLMs, while powerful, often require large datasets and significant computational resources, posing challenges for low-resource languages. In contrast, the Bi-Directional Head-Driven Parser offers valuable advantages for specific applications with limited datasets and provides precise syntactic understanding, crucial for tasks such as machine translation.

Overall, our findings reflect a significant step forward in the field of NLP parsing techniques. The Bi-Directional Head-Driven Parser not only advances our computational understanding of human languages but also opens avenues for more targeted and effective language processing applications. As the field continues to evolve, integrating these advancements will be essential for addressing the diverse challenges of natural language processing and achieving more refined language technologies.

REFERENCES

[1] Joshi, A. K. (1985). Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions? In Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics (pp. 154-160).

[2] Vijay-Shanker, K., & Weir, D. J. (1994). The equivalence of four extensions of context-free grammars. Mathematical Systems Theory, 27(2), 101-120.

[3] Kurariya, P., Chaudhary, P., Jain, P., Lele, A., Kumar, A., & Darbari, H. (2015, September). File model approach to optimize the performance of Tree Adjoining Grammar based Machine Translation. In 2015 International Conference on Computer, Communication and Control (IC4) (pp. 1-6). IEEE.

[4] Kurariya, P., Chaudhary, P., Bodhankar, J., Singh, L., Kumar, A., & Darbari, H. (2020, December). TREE ADJOINING GRAMMAR BASED “LANGUAGE INDEPENDENT GENERATOR”. In Proceedings of the 17th International Conference on Natural Language Processing (ICON) (pp. 138-143).

[5] Kurariya, P., Chaudhary, P., Bodhankar, J., Singh, L., & Kumar, A. (2024, August). "BI-Directional Head-Driven Parsing for English to Indian Languages Machine Translation”. In Proceedings of the 4th International Conference on NLP & Data Mining (pp. 71-81).

[6] Kurariya, P., Chaudhary, P., Bodhankar, J., Singh, L., Kumar, A., & Darbari, H. (2022, October). VTAG: Virtual Lab for Tree-Adjoining Grammar-Based Research. In International Conference on Information and Communication Technology for Competitive Strategies (pp. 765-777). Singapore: Springer Nature Singapore.

[7] Kurariya, P., Chaudhary, P., Bodhankar, J., Singh, L., & Kumar, A. (2023, August). Unveiling the Power of TAG Using Statistical Parsing for Natural Languages. In CS & IT Conference Proceedings (Vol. 13, No. 14). CS & IT Conference Proceedings.

[8] Chomsky, N. (1956). Three models for the description of language. IRE Transactions on Information Theory, 2(3), 113-124.

[9] Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of the Eighteenth International Conference on Machine Learning (pp. 282-289).

[10] Eisner, J. (2012). Three new probabilistic models for dependency parsing: An exploration. In Proceedings of the 16th Conference on Computational Natural Language Learning (pp. 25-36).

[11] Hale, J. (2001). A probabilistic Earley parser as a psycholinguistic model. In Proceedings of the North American Chapter of the Association for Computational Linguistics Conference (pp. 159-166).

[12] Joshi, A., Levy, L. S., & Takahashi, M. (1992). Tree adjoining grammars. In A. von Stechow & D. Wunderlich (Eds.), Handbook of Contemporary Syntactic Theory (pp. 65-130). Berlin, Germany: De Gruyter Mouton.

[13] Satta, G., & Stock, O. (1994). Bidirectional context-free grammar parsing for natural language processing. Artificial Intelligence, 69(1-2), 123-164.

[14] Zhou, J., & Zhao, H. (2019). Head-driven phrase structure grammar parsing on Penn treebank. arXiv preprint arXiv:1907.02684.

[15] Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to Information Retrieval. Cambridge University Press.

AUTHORS

Mr. Pavan Kurariya is a Scientist 'E' working in the HPC AI-IT Infra. & Operations Group at C-DAC Pune and has more than 15 years of experience. He is a distinguished researcher, and his expertise lies in various domains such as Natural Language Processing, Cyber Security, Cryptography, and Quantum Computing. He has contributed significantly to the advancements of Machine Translation, Cyber Security, and Quantum Computing. His primary area of interest centers around Machine Translation and Cryptography, where he investigates novel techniques and cutting-edge methodologies to enhance the accuracy and efficiency of various NLP applications.

Mr. Prashant Chaudhary is a Scientist 'E' working in the AAI & GIST Group at C-DAC Pune and has more than 15 years of experience. He is a distinguished researcher, and his expertise lies in various domains such as Natural Language Processing, Machine Translation, and Cyber Security. Through his numerous research papers, he has made significant contributions to the field of Machine Translation by investigating both theoretical aspects and practical applications. His primary area of interest centers around Natural Language Processing, where he investigates cutting-edge techniques and methodologies to enhance the accuracy and efficiency of various NLP applications.

Ms. Jahnavi Bodhankar is a Scientist ‘F’ working in the HPC AI-IT Infra. & Operations Group at C-DAC Pune and has more than 18 years of experience. She is a distinguished researcher, and her expertise lies in various domains such as Natural Language Processing, Cyber Security, Machine Learning, and Blockchain Technology. She has contributed significantly to the advancements and understanding of NLP, E-Signature, and Blockchain through her numerous research papers and intricate work.

Ms. Lenali Singh is a Scientist ‘F’ working in the AAI & GIST Group at C-DAC Pune and has more than 20 years of experience. Her key role is in initiating and executing various projects in the areas of Natural Language Processing and Speech Technology. She is a distinguished researcher, and her expertise lies in various domains such as Natural Language Processing and Speech Technology. She has contributed significantly to the advancements and understanding of the NLP field through her numerous research papers and intricate work.

Dr. Ajai Kumar is a Scientist ‘G’ and Head of the AAI & GIST Group at C-DAC Pune, with more than 20 years of experience working in Natural Language Processing, including Machine Translation, Speech Technology, Information Extraction & Retrieval, and E-learning systems. His key role is in initiating mission mode consortium projects in the areas of Natural Language Processing, Speech Technology, Video Surveillance, etc. Through his meticulous research, he aims to bridge the gap between different languages and enable seamless communication across linguistic boundaries.

**** C-DAC**: Centre for Development of Advanced Computing is the premier R&D organization of the Ministry of Electronics and Information Technology (MeitY), Government of India*