**CFG To PDA Converter**

1. Sanika Kadam; 12010468; AI/A; 53

2. Omkar Kale; 12010422; AI/A; 54

3. Anish Kamble; 12011133; AI/A; 55

4. Aditya Kanawade; 12010940; AI/A; 58

5. Sejal Kharche; 12010585; AI/A; 63

**Introduction**

A context-free grammar is a simple and exact technique for defining how phrases in languages are constructed from smaller blocks, capturing the natural "block structure" of sentences. The formalism's simplicity makes it open to rigorous mathematical analysis. Context-free grammars are also simple enough to allow pushdown automata to be used to build efficient parsing algorithms (PDAs). These "predicting machines" employ information about the contents of their stack to decide whether or not a given string can be generated by the grammar. The power of a CFG and a PDA is the same: a CFG generates context-free language, while a PDA recognises context-free language. If some pushdown automata recognises a language, it is context-free. Because of this equivalence, a CFG can be used to specify a programming language, and the equivalent PDA can be used to construct the language's compiler.

**Literature Survey**

In this paper[1], we investigate the topic of learning context-free grammar from a corpus in this paper. We look into a technique based on the concept of the corpus's minimum description length. When compared to the initial cost, the overall cost, or description length, drops by as much as 14%.

[2] recognises the components of a source code that need to be updated and generates new code segments using context-free grammars. Furthermore, because parsing produces larger and less ambiguous meaning-bearing structures in the sentence, the design and implementation of CFG can be critical to the performance of high-level natural language processing tasks like question answering [3] and interactive voice response [4] systems.

Push-down automata (PDA) are nondeterministic machines. They have a stack on which they store information in a last-in-first-out fashion. PDA's can push a symbol onto the stack, move its read head one cell to the right and enter a new state. Because the PDA is nondeterministic, many transitions may be conceivable in a given configuration. The class of context free languages, which strictly includes the class of regular languages (accepted by finite state automata) and is strictly contained in the class of recursive enumerable languages, is the class of languages accepted by PDAs (accepted by Turing machines).

**Problem Statement**

How to convert a CFG into a PDA that recognizes the language specified by the CFG?

**Proposed system**

In this system, we demonstrate how Context-free grammar converts to pushdown automaton with string parser works with Greibach Normal Form inputs.

**Conclusion**

We have demonstrated that pushdown automata can detect the context-free language category. A finite automaton recognises every regular language. Every finite automaton is a stack-agnostic pushdown automaton. Every regular language is devoid of context.

**References**

[1] Automatic Learning of Context-Free Grammar

[2] Mark van den Brand, Alex Sellink, and Chris Verhoef, “Generation of components for software renovation factories from context-free grammars”, In Working Conference on Reverse Engineering, IEEE Computer Society, WCRE97, pp. 144-153.

[3] C. Yuan and C. Wang, “Parsing model for answer extraction in Chinese question answering system”, Proceedings of IEEE NLP-KE ’05, pp. 238 - 243.

[4] M. Balakrishna, D. Moldovan, E.K. Cave, “Automatic creation and tuning of context free grammars for interactive voice response systems”, Proceedings of IEEE NLP-KE ’05, pp. 158 - 163.

[5] Autonomous push-down automaton built on DNA, Tadeusz KRASIŃSKI , Sebastian SAKOWSKI , Tomasz POPŁAWSKI