Constraint Programming Algorithms in Generative AI: Enhancing Fast, Rule-Compliant Solutions

Shishir Biyyala

October 26th, 2024

Outline

Introduction

Shishir Biyyala

- Accomplished software engineering leader driving innovation and delivering complex projects.
- `https://www.linkedin.com/in/sbiyyala/`

AI and Fast, Rule-Compliant Systems

- Recent AI adoption is driven by the need for cost reduction, automation, and more accessible AI tools.
- 42% of enterprise companies have deployed AI, while 40% are exploring or experimenting.
- Generative AI is becoming prominent due to advancements in ease of use and scalability.
- Financial services and telecommunications lead in AI adoption, driven by operational optimizations.
- Overcoming barriers like limited skills and data complexity is critical for successful AI deployment.

Constraint Programming (CP)

What is Constraint Programming?

- CP is a declarative paradigm used to solve combinatorial problems
- It involves specifying variables, constraints, and use of search algorithms to find solutions
- Sub-field of AI that intersects with optimization and decision-making
- It ensures AI-generated solutions are rule-compliant, which is crucial in regulated industries like finance and healthcare.
- CP is used in logistics optimization, route planning, operations management, and resource allocation.

Role of CP in Generative AI

- CP ensures that AI-generated solutions satisfy specific rules and conditions, crucial for compliance-heavy industries.
- CP is vital for maintaining compliance in generative models, allowing AI to generate outputs that align with strict requirements.

CP Techniques in Generative AI

- Constraint Satisfaction Problems (CSPs) and Constraint Optimization Problems (COPs)
  - CSPs involve finding any solution that satisfies all constraints, while COPs involve finding the best solution based on an objective function.
- Example Techniques
  - Backtracking tries and eliminates options that don't meet constraints.
  - Local Search iteratively improves a solution by making small adjustments.
  - Constraint Propagation reduces search space by removing invalid values early.

Constraint Propagation

- Constraint Propagation is a key CP technique used to reduce the search space.
- It removes invalid possibilities early in the search, leading to faster solutions.
- Relevance to Generative AI: Helps AI systems make structured decisions in real-time.

Constraint Propagation example

Conference Scheduling

- Schedule 50,000 conference talks across multiple rooms such that no two talks assigned to the same room overlap in time.
- Naive approach: Uses a brute-force algorithm to try all possible room assignments.
- Constraint propagation: Sorts talks by start times to reduce conflict-checking complexity, then assigns talks by end times to minimize comparisons and reduce execution time.
- `https://github.com/nvta-sbiyyala/icsiscet-talk/tree/main/src`

Conclusion + Q&A

Conclusion

- CP is a powerful tool for solving complex problems in AI
- Constraint Propagation allows for rapid, efficient query handling in Generative AI
- Explore more use cases and applications of CP in intelligent systems!
- Talk artifacts available in `https://github.com/nvta-sbiyyala/icsiscet-talk`