LLMASP

Synthesising Answer Set Programs with LLMs

JESSE HEYNICK | KEEGAN O'BRIEN | ROWAN ROSENBERG | JULYAN VAN DER WESTHUIZEN

We use Large Language Models to translate natural language into Answer Set Programs, testing fine-tuning, prompting, and solver feedback to overcome the knowledge acquisition bottleneck.

1 Answer set programming

ASP is a declarative programming paradigm where problems are encoded as logical rules.

```
giftedChild.
greyHair.
studiedMusic :- giftedChild.
bach :- studiedMusic, greyHair, not beethoven.
composed(X) :- studiedMusic, talent(X).
talent(f(bach)).
```

2 Problem statement and motivation

Writing ASP by hand is slow, expert-dependent, and creates a Knowledge Acquisition Bottleneck, with no effective tools for generating ASP from natural language and only limited prior attempts.

3 We evaluate 3 LLM approaches in solving the Knowledge Acquisition Bottleneck.

Fine-Tuning

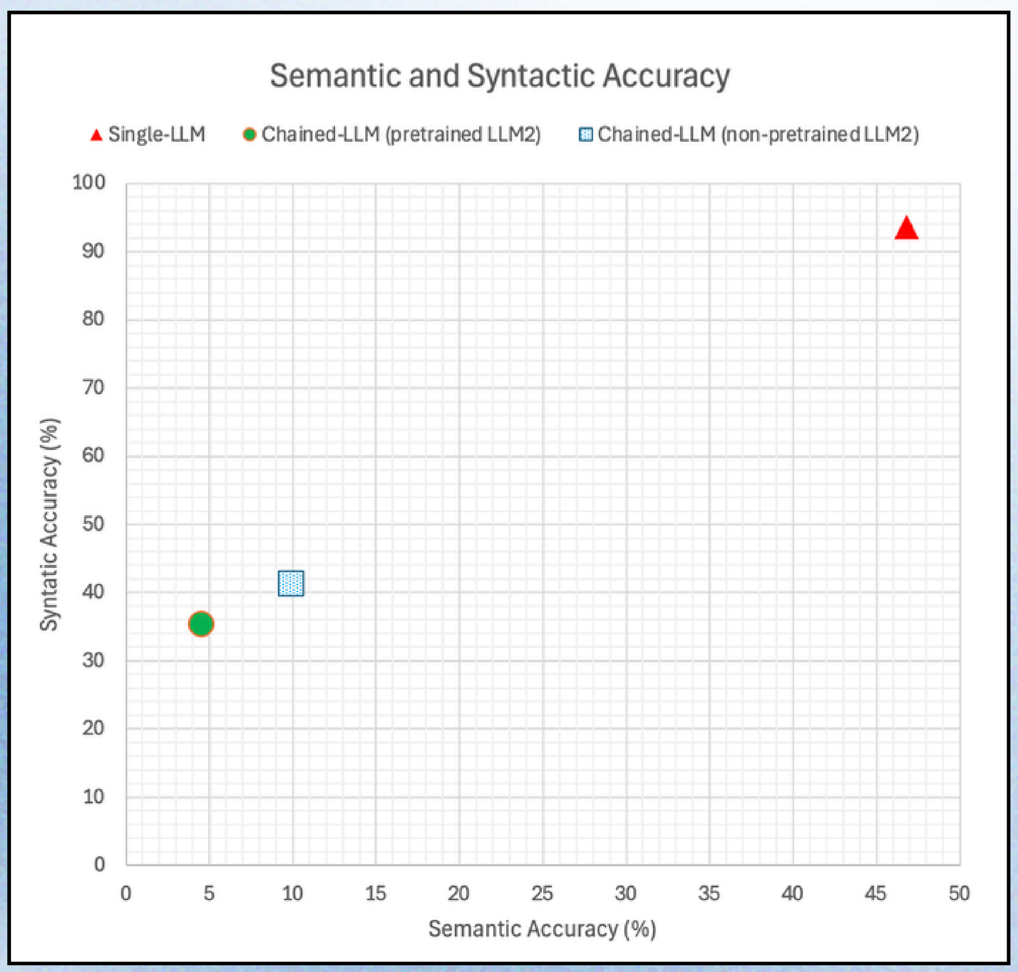
- **Methodology**: Fine-tuned single and chained LLMs on synthetic ASP data.
- **Results**: Single LLMs excel; chained models lose accuracy.

COT Prompting

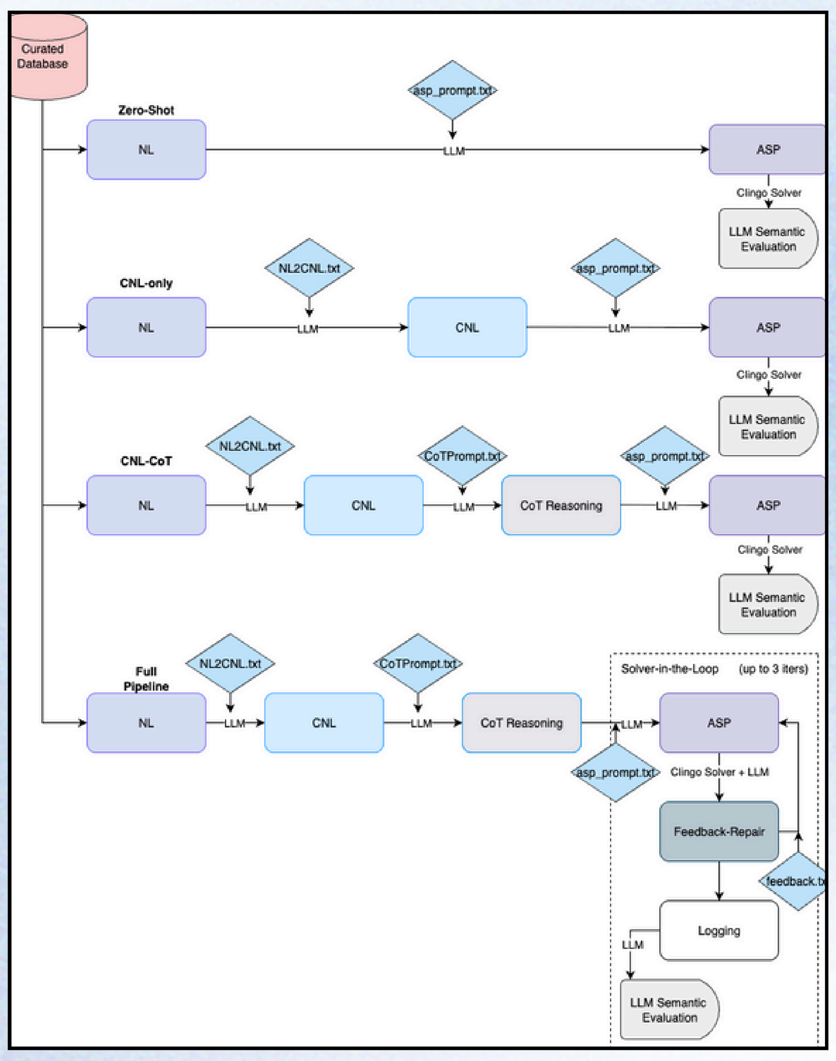
- **Methodology**: Chained LLM pipeline with solver-in-the-loop, tested on 27 tasks.
- **Results**: GPT-5 near-perfect; GPT-4o lags behind.

Few-Shot Prompting

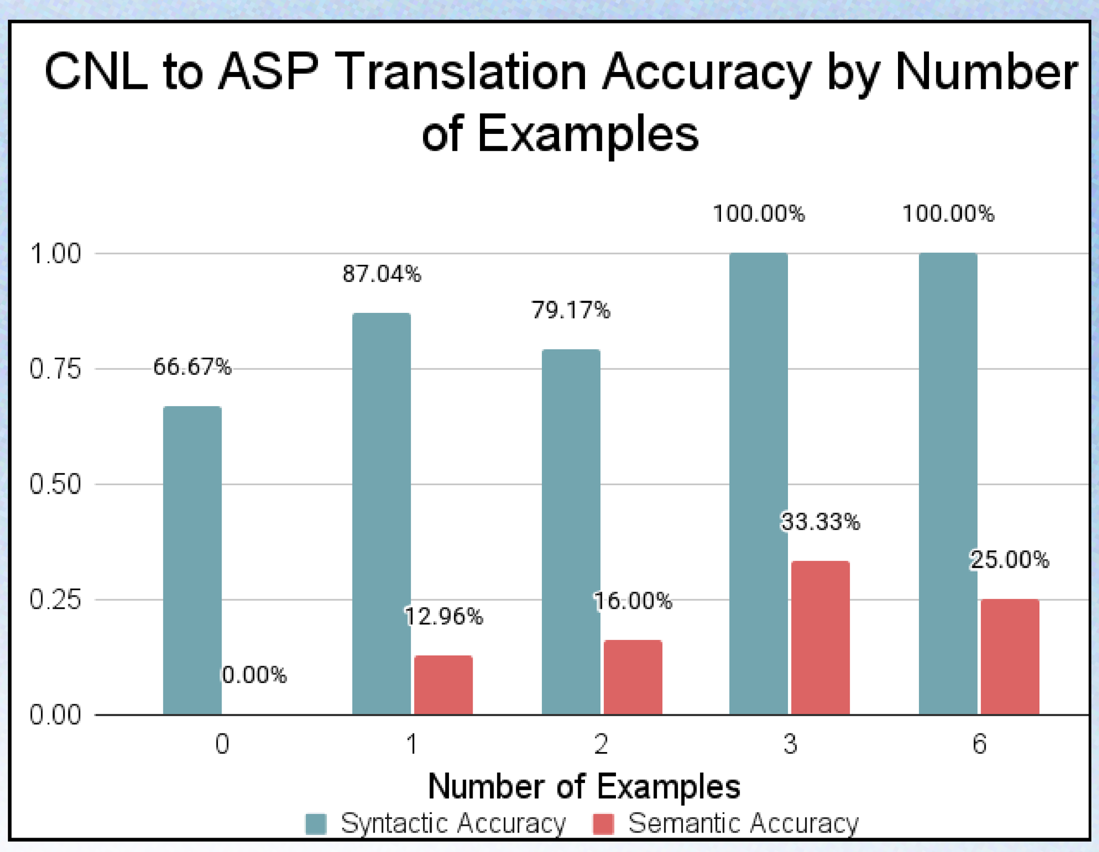
- **Methodology**: Appending gold-standard, input-output pairs to the prompt
- **Results**: Best results from 3 examples of same difficulty and mixed domain



Syntactic and Semantic Accuracy of Single-LLM systems vs Chained-LLM systems



Architecture Diagram of various COT pipeline approaches



Syntactic and Semantic Accuracy by number of Few-Shot examples

4 Impact and Future Work and Conclusion

LLMs can ease the knowledge bottleneck in ASP, making it more accessible and accurate. Future work will expand datasets, improve generalisation, and refine pipelines for real-world use.

University of Cape Town

School of IT