MONASH University

Formal Explainability for Artificial Intelligence in Dynamic Environments

Jaime Cuartas Granada

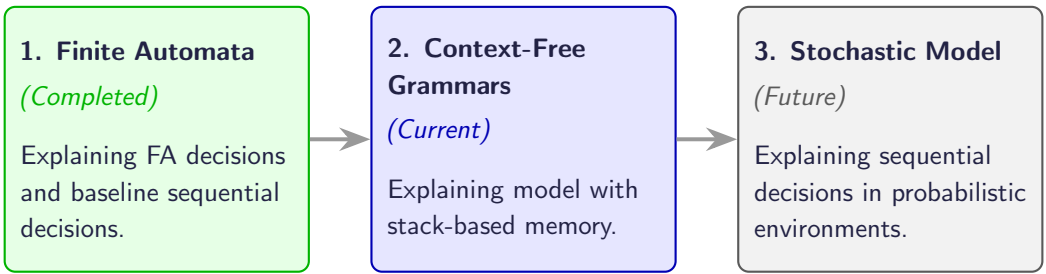
26-February-2026

Department of Data Science and Artificial Intelligence (DSAI), Monash University, Australia

Project Summary & Refinements

Summary

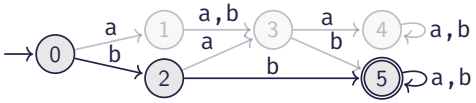
Goal: Deliver explanations for sequential decision-making models.



Completed Work: Finite Automata

Explaining Finite Automata (Completed)

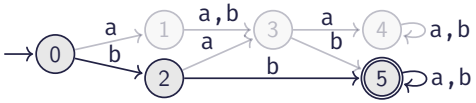
- FA is a mapping from an input $w \in \Sigma^*$ to a class $\mathcal{K} = \{\text{Accept}, \text{Reject}\}$.



Input w**:**	b b b b b	$\rightarrow$ Accept
AXp 1:	b b Σ Σ Σ	$\rightarrow$ Guarantees Accept $L(\text{bb}\Sigma\Sigma\Sigma) \subseteq L(\mathcal{A})$
AXp 2:	Σ Σ b Σ Σ	$\rightarrow$ Guarantees Accept $L(\Sigma\Sigma\text{b}\Sigma\Sigma) \subseteq L(\mathcal{A})$

Explaining Finite Automata (Completed)

- FA is a mapping from an input $w \in \Sigma^*$ to a class $\mathcal{K} = \{\text{Accept}, \text{Reject}\}$.



Input w**:**	b b b b b	$\rightarrow$ Accept
CXp 1:	Σ b Σ b b	$\rightarrow$ Enables Reject $L(\Sigma\texttt{b}\Sigma\texttt{bb}) \not\subseteq L(\mathcal{A})$
CXp 2:	b Σ Σ b b	$\rightarrow$ Enables Reject $L(\texttt{b}\Sigma\Sigma\texttt{bb}) \not\subseteq L(\mathcal{A})$

Why Finite Automata Are Not Enough

- $L = \{a^n b^n \mid n \geq 1\}$ cannot be described by a FA.
- For a rejected word like aaaabb, standard parsers identify the error at the end.

Source Code:

```
1  int main(){
2      for(int i=0; i<10; i++){{  // Error
3          printf("hello");
4      }
5  }
```

Parser Output:

```
error: expected '}' at end of input
5 | }
  |  ^
```

Current Research: Explaining CFGs

Explaining Context-Free Grammars

- Transition to Context-Free Grammars (CFGs).
- Consider $G = (\{B\}, \{(,)\}, R, B)$ for balanced parentheses, with rules R:

$$\begin{aligned} B &\to (\; B \;) \; B \quad \text{(Rule 1)} \\ B &\to (\; B \;) \quad \text{(Rule 2)} \\ B &\to (\;) \; B \quad \text{(Rule 3)} \\ B &\to (\;) \quad \text{(Rule 4)} \end{aligned} \tag{1}$$

Framing the CFG Problem

- **The New Classifier:** We frame Context-Free Grammar (CFG) membership as our decision model.
- For a grammar G and an input word w:
  - **Accept:** $w \in L(G)$ (Valid syntax)
  - **Reject:** $w \notin L(G)$ (Syntax error)

- **The Explanation Goal:**
  - Extract an **AXp**: The minimal subset of tokens that guarantees the current syntax status.
  - Extract a **CXp**: The minimal subset of tokens to free (replace with Σ) to flip the syntax status.

Fragility of Acceptance vs. Robustness of Rejection

In Context-Free Languages (like balanced parentheses):

Accepted Words are Fragile:

- $w = ()()$
- `color()`
- text

Rejected Words are Robust:

- $w =))))$
- $((\,)), \,()(\,)$

Challenge: How do we formally extract these complex, coordinated CXps for rejected words?

Methodology: The Modified CYK Algorithm

- To verify if a set of indices S is a valid CXp, we must check if replacing those tokens with wildcards (Σ) allows the word to be accepted.
- We achieve this by modifying the base case of the **CYK Parsing Algorithm**:

Modified Base Case Initialization

For each token w_i at index i:

- **If $i \notin S$ (Fixed):** $T[i, i] = \{A \in V \mid A \to w_i\}$
- **If $i \in S$ (Freed/Wildcard):** $T[i, i] = \{A \in V \mid \exists \alpha \in \Sigma, A \to \alpha\}$

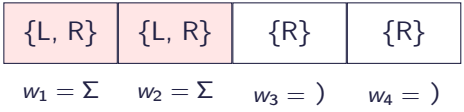
Result: If the Start symbol S appears at the top of the table $T[1, n]$, then the candidate set S is a valid CXp.

Visualizing Extraction: Greedy Deletion

Algorithm Insight: We start by freeing the *entire* word, then greedily lock tokens back into place one by one. If locking a token breaks the parsing, it belongs in our minimal CXp.

Example: $w =$))))), checking CXp candidate $S = \{1, 2\}$

Wildcards Fixed indices lock to $R \to)$



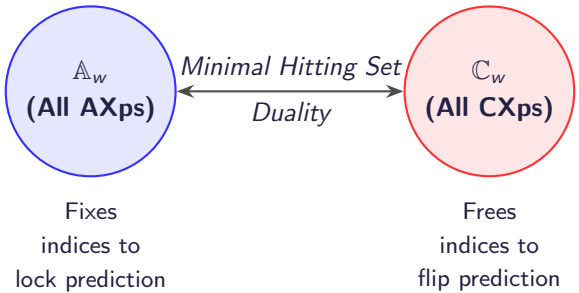
Because indices 1 and 2 can now act as L ($L \to ($), the CYK table successfully resolves to the Start variable, proving $\{1, 2\}$ is a valid explanation.

Current Status: CFG Explanations

- **Theoretical Foundation:**
  - Formalized the extension of AXps/CXps to Context-Free Languages.
  - Proved the correctness of the Modified CYK approach for wildcard evaluation.
- **Algorithmic Implementation:**
  - Currently implementing the greedy extraction algorithm (ExtractCXp).
  - Testing extraction performance on standard balanced parenthesis (Dyck) languages.
- **Next Immediate Step:** Benchmarking the computational complexity of the CYK-based extraction and drafting the manuscript for this second phase of the thesis.

The Minimal Hitting Set Duality

- **The Core Relationship:** Abductive and contrastive explanations share a formal duality [?].
- Every AXp is a minimal hitting set of the complete set of CXps, and vice versa.
- To flip a prediction (CXp), you must free at least one token from every reason that guarantees the current prediction (AXp).



Outcomes: Enumeration & Milestone

- **Algorithmic Contribution:**
  - Leveraged this duality to develop algorithms for the formal **enumeration** of explanations in Finite Automata.
  - Successfully maps the abstract concepts of XAI onto rigorous formal language properties.

Phase 1 Milestone Achieved

Status: Completed.

Output: The formal definitions, duality proofs, and enumeration algorithms have been compiled and submitted to **ICALP 2026**.

- *Transitioning to Phase 2:* With the baseline for regular languages established, we now scale the complexity to languages requiring memory.

Motivation

- ADD CONTENT

Separated content, e.g., variable explanations

Frame without a title

- Frames do not need to have a title...

Math Expressions

Integrals and Other Expressions

$$\iint_{\partial\Omega} f(x)\mathrm{d}x \in \mathbb{C} \tag{2}$$

$$E = mc^2 \tag{3}$$

$$F = ma \tag{4}$$

m Mass

c Speed of light

Theorem

The following statement is correct

$$\frac{\partial f(\vec{x})}{\partial x_i} = \sum_{l=1}^{L} \cos\left(l\frac{2\pi}{L} + 0\right) \tag{5}$$

Elements

Typography

```
The theme provides sensible defaults to
^^I^^I\emph{emphasize} text, \alert{accent} parts
^^I^^Ior show \textbf{bold} results.
```

becomes

The theme provides sensible defaults to *emphasize* text, **accent** parts or show **bold** results.

Font feature test

- Regular
- *Italic*
- Small Caps
- **Bold**
- ***Bold Italic***
- **Bold Small Caps**
- `Monospace`
- *`Monospace Italic`*
- **`Monospace Bold`**
- ***`Monospace Bold Italic`***

Lists

Items

- Milk
- Eggs
- Potatoes

Enumerations

1. First,
2. Second and
3. Last.

Descriptions

PowerPoint Meeh.

Beamer Yeeeha.

Tables

Table 1: Largest cities in the world (source: Wikipedia)

City	Population
Mexico City	20,116,842
Shanghai	19,210,000
Peking	15,796,450
Istanbul	14,160,467

Blocks

Three different block environments are pre-defined and may be styled with an optional background color.

Default

Block content.

Alert

Block content.

Example

Block content.

Default

Block content.

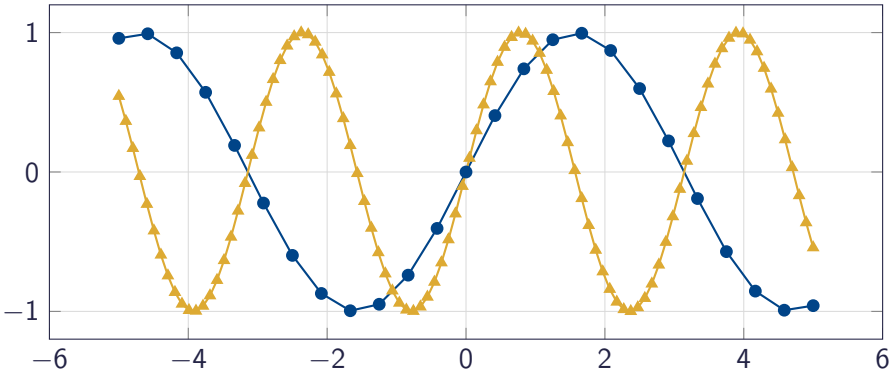
Alert

Block content.

Example

Block content.

Line plots



Standout Frame!

Backup slides

Sometimes, it is useful to add slides at the end of your presentation to refer to during audience questions.

The best way to do this is to include the `appendixnumberbeamer` package in your preamble and call `\appendix` before your backup slides.

The theme will automatically turn off slide numbering and progress bars for slides in the appendix.