Legal Citation Machine

AI-powered precedent mapping and citation-strength analysis with human-style reasoning

December 10th, 2025

UC Berkeley School of Information

Agenda

01	Problem Space
02	Capabilities & Prototype
03	Technical Approaches
04	Knowledge Graph
05	Citation Classifier
06	Case Classifier
07	Conclusion

Team Members

Simran Gill
Product Manager
& Data Scientist

Kent Bourgoing
Machine Learning / AI Engineer

Bryan Guan
Data Engineer

Wendy Tian
Project Manager
& Data Scientist

Hunter Tonn
UX Designer
& Data Scientist

Problem Space

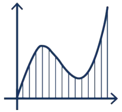
Problem Statement, Existing Solutions, Users and Impact

Problem

Lawyers **rely on citations** to build arguments and persuade courts

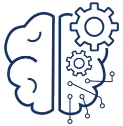
… But today's citator tools are often **proprietary, expensive, and built on dated infrastructure.**

Impact and Opportunity



Targets a **fast-growing, multi-billion-dollar legal tech and research market** where firms are actively adopting AI.

Targets the legal research market, where major platforms like Westlaw and LexisNexis serve **thousands of law firms and generate billions annually**



Real value is in demonstrating that **LLM-based systems can deliver citation analysis with the same (or better) accuracy, and transparency** as traditional tools.

Problem Space

What Makes this Different?

Existing Solutions	WK <> Berkeley Solution
Unstructured Search Results Long keyword lists force researchers to open case after case to find useful facts or context.	**Structured, scannable snapshots** Single search present critical details in a structured layout.
Shallow treatment labels Fixed tags like “followed” don’t capture the court’s actual reasoning.	**Human-like rationales** LLM explains how later courts relied on or limited the case with paragraph-level reasoning.
Hard-to-read citation safety Determining whether a case is safe to cite requires lengthy manual effort.	**Transparent citation-strength** Simple formula turns full treatment history into a single, reliable signal.
Complex workflows Tools require manual digging and their interfaces take too long to master	**Intuitive interface** Key information surfaces instantly with evidence visible for quick validation.

The Value of Our Solution

Accurate citation analysis that **lawyers can rely on.** Our system demonstrates that modern LLMs can serve the same purpose as traditional citators while showing reasonable rationale.

Capabilities & Prototype

Capabilities, Demo

Interactive Case Lookup & Chatbot Interface

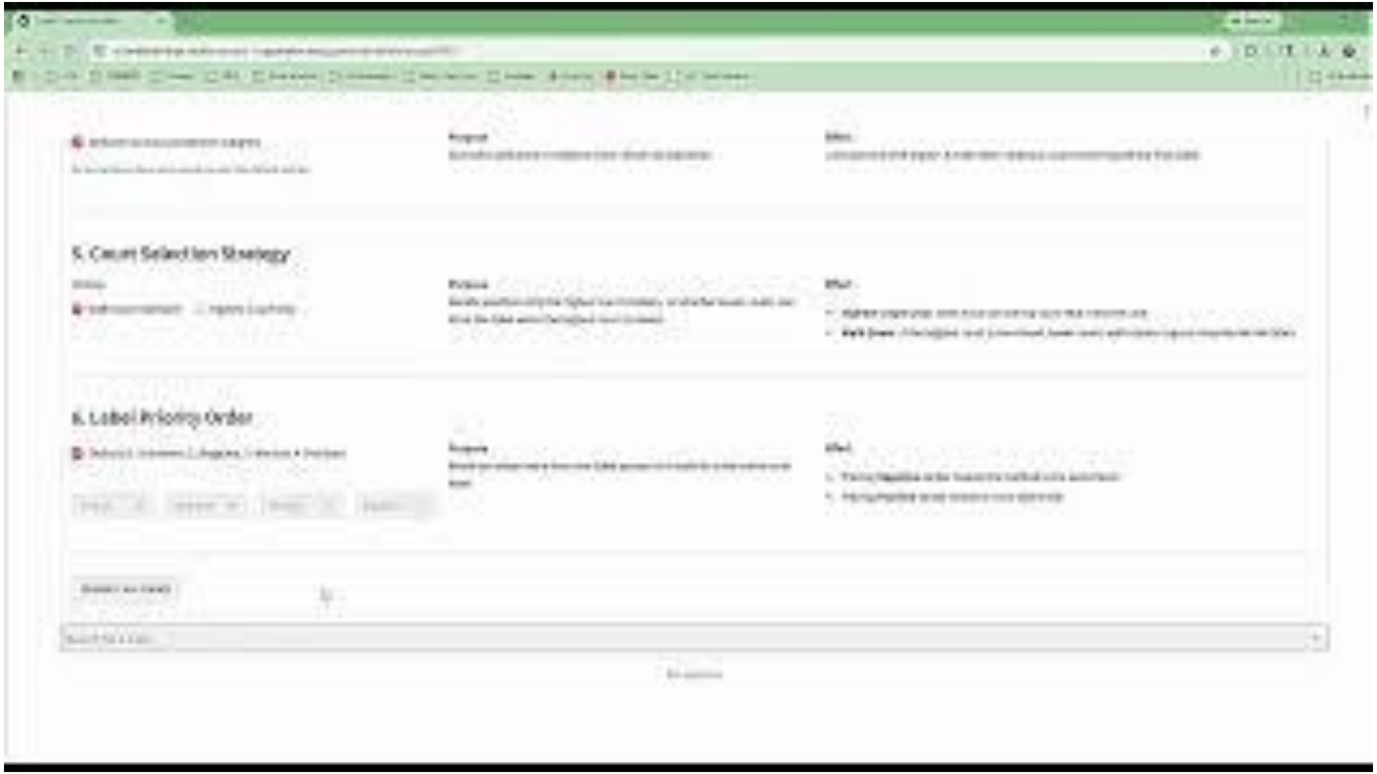
Case Lookup

- **Easy-to-use**, targeted search across cases
- Immediate case **classification with rationale**
- Key case metadata
- Full list of citing cases with **treatment labels and rational**, exportable to CSV

Chatbot

- Ask any **ADA-related questions**, see citation history, and surface related facts stored in Neo4j
- Explore **citation patterns and precedent**
- **Compare** cases and treatments
- Get step-by-step **reasoning** in plain English

Demo



Chatbot – Types of Questions You Can Ask

Orientation: *“What kinds of cases are in your database?”*

Single case: *“Summarize Access Now v. Southwest Airlines.”*

Citation treatment: *“Show citing cases that criticize Access Now and explain how.”*

Compare cases: *“Compare [Case A] and [Case B] on ‘major life activity’.”*

ADA concepts: *“What is a ‘qualified individual with a disability’ under the ADA?”*

Scenarios: *“Given this fact pattern, which ‘Good’ precedents support the employee?”*

Patterns / research: *“Show ADA cases on remote work as a reasonable accommodation.”*

Technical Approaches

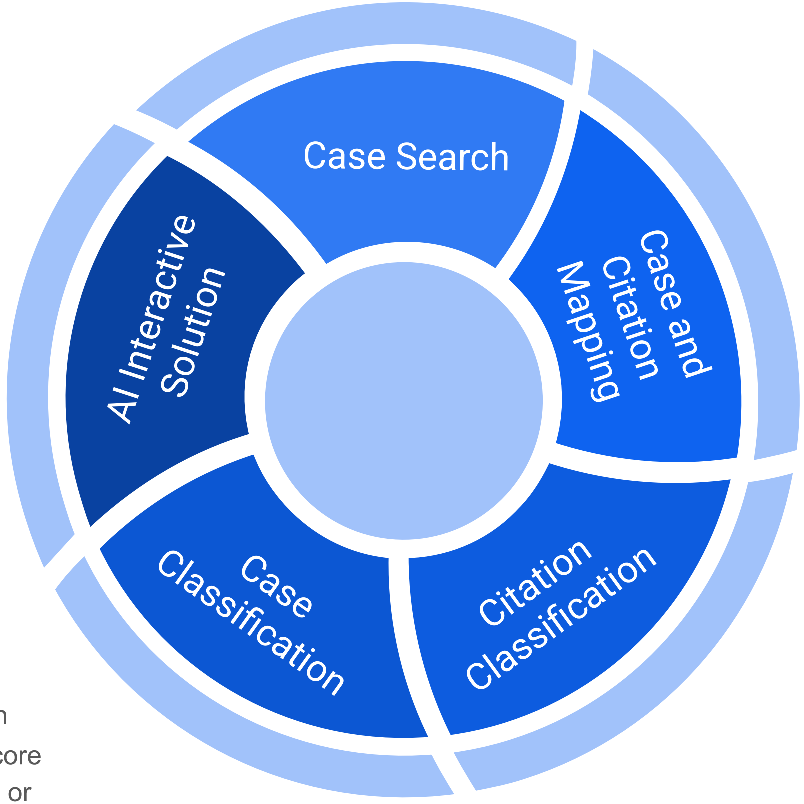
Technical Approaches

Capabilities

HOW TO UNDERSTAND THE CITED CASE QUICKLY? Find any case fast and surface key facts (court, date, summary, and link) at a glance.

HOW TO UNDERSTAND THE LABEL QUICKLY

Visualize the case information, citation relationship and case labels, and interact with user to deliver a comprehensive solution.



HOW TO CONNECT CASES AND CITATIONS? Visualize every citing case in a clean, structured map that's easy to scan.

HOW DOES THE CITING CASE TREAT THE CITED CASES? Label each citation as positive, neutral, or negative with paragraph-level reasoning that tracks real legal analysis.

HOW IS THE CITED CASE TREATED OVERALL BY THE CITING CASES? Roll up all citation treatments into a clear, formula-driven score showing whether a case is strong, weak, or mixed precedent.

Technical Approaches

A 4-Step Process to Realize the Capabilities

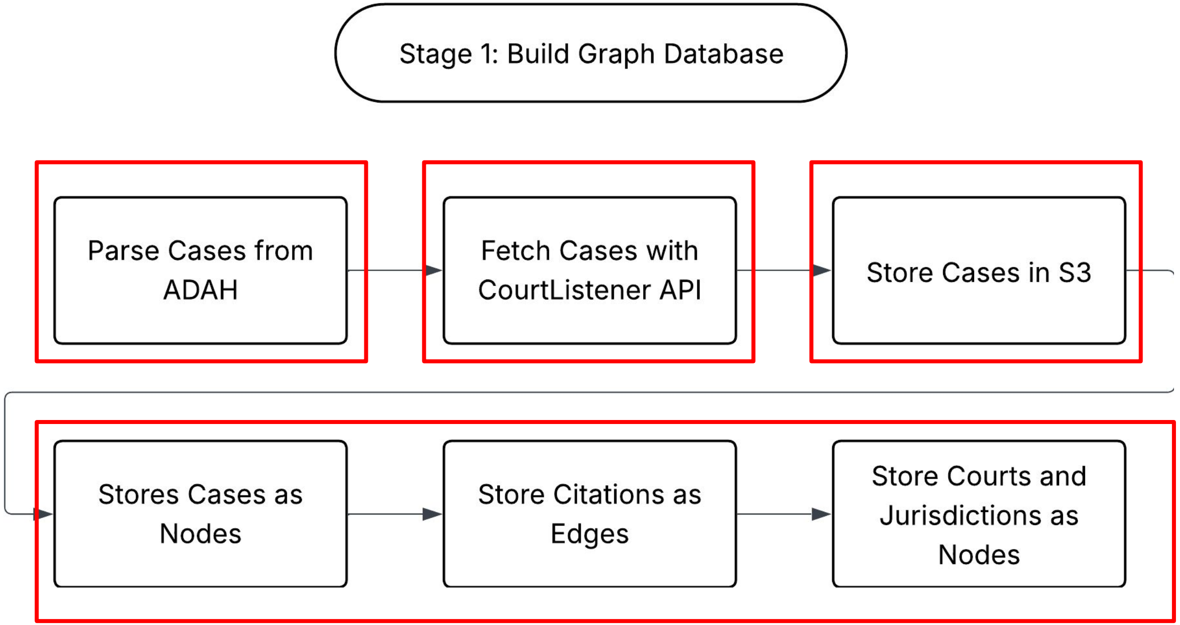
Step 1: Develop a semantic layer of cases, using knowledge graph to capture attributes such as name, decision date, court, jurisdiction, summary, and URL

Step 2: Develop classifier for each single citation

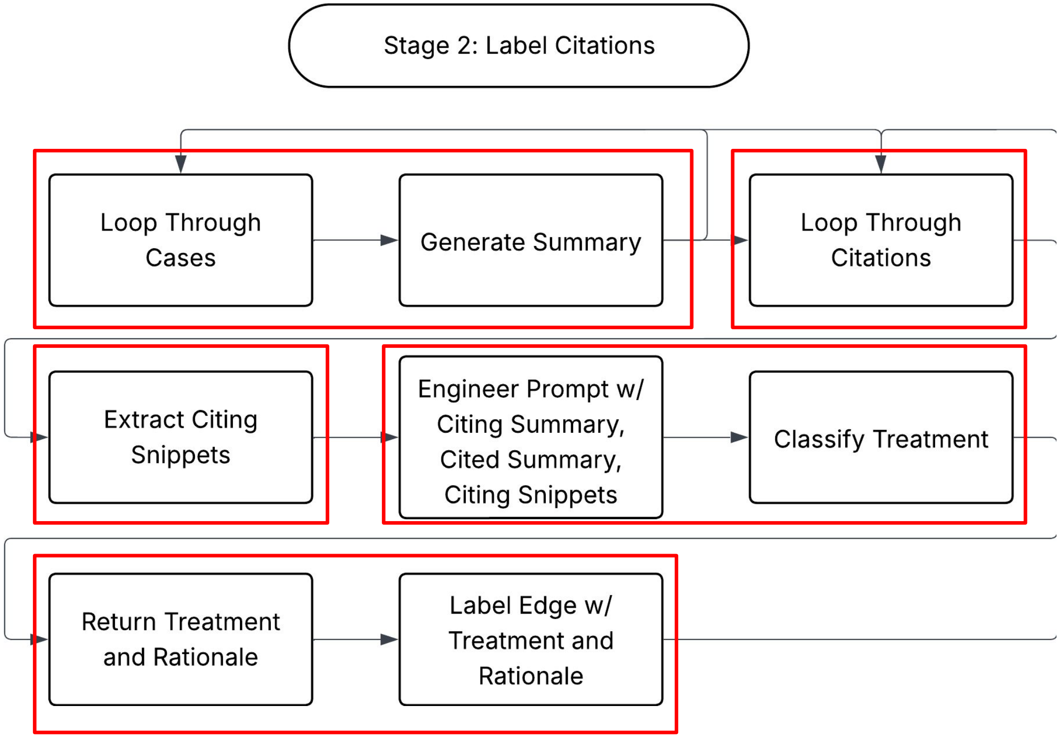
Step 3: Develop computational algorithm to classify individual case Label the Case

Step 4: Deploy to end user with interactive search machine and chatbot design

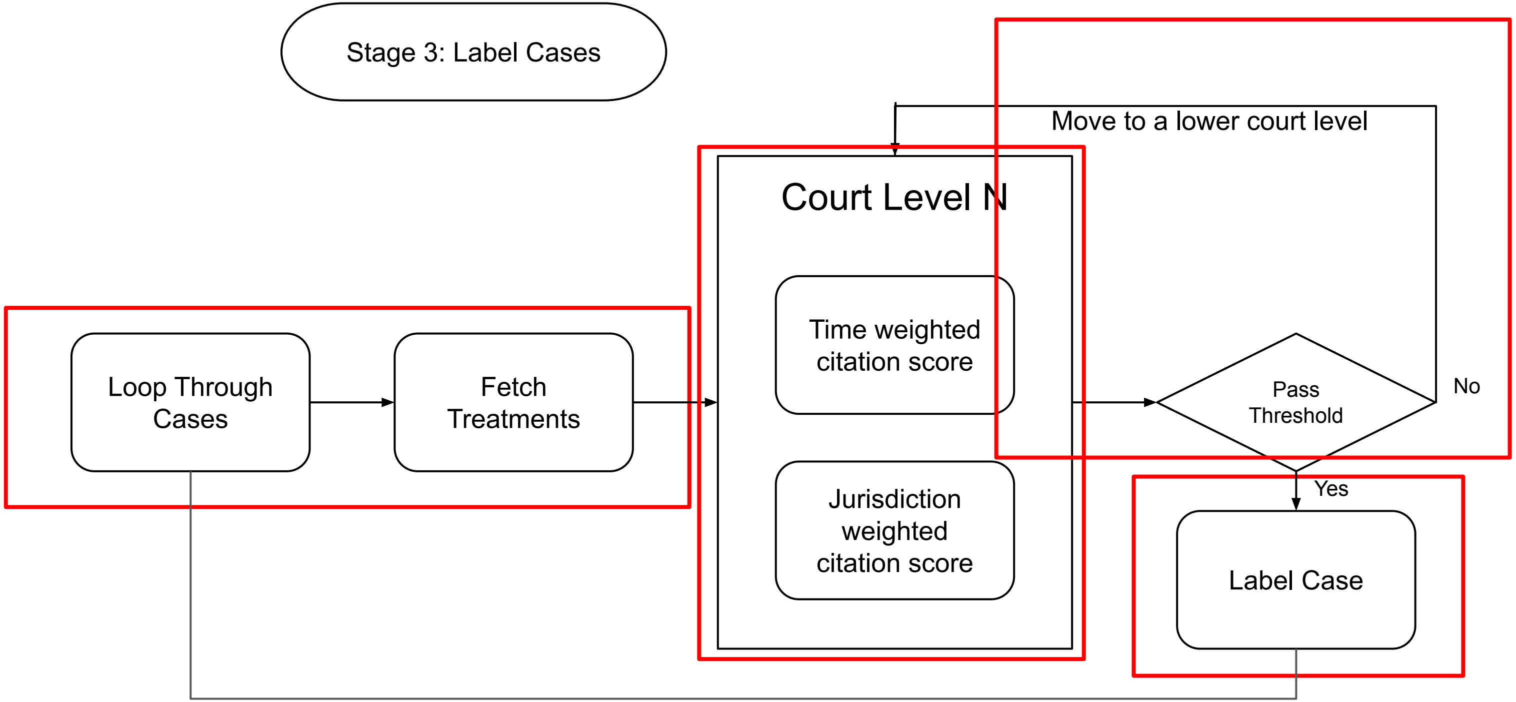
Stage 1: Build a Graph Database for Case Search and Mapping



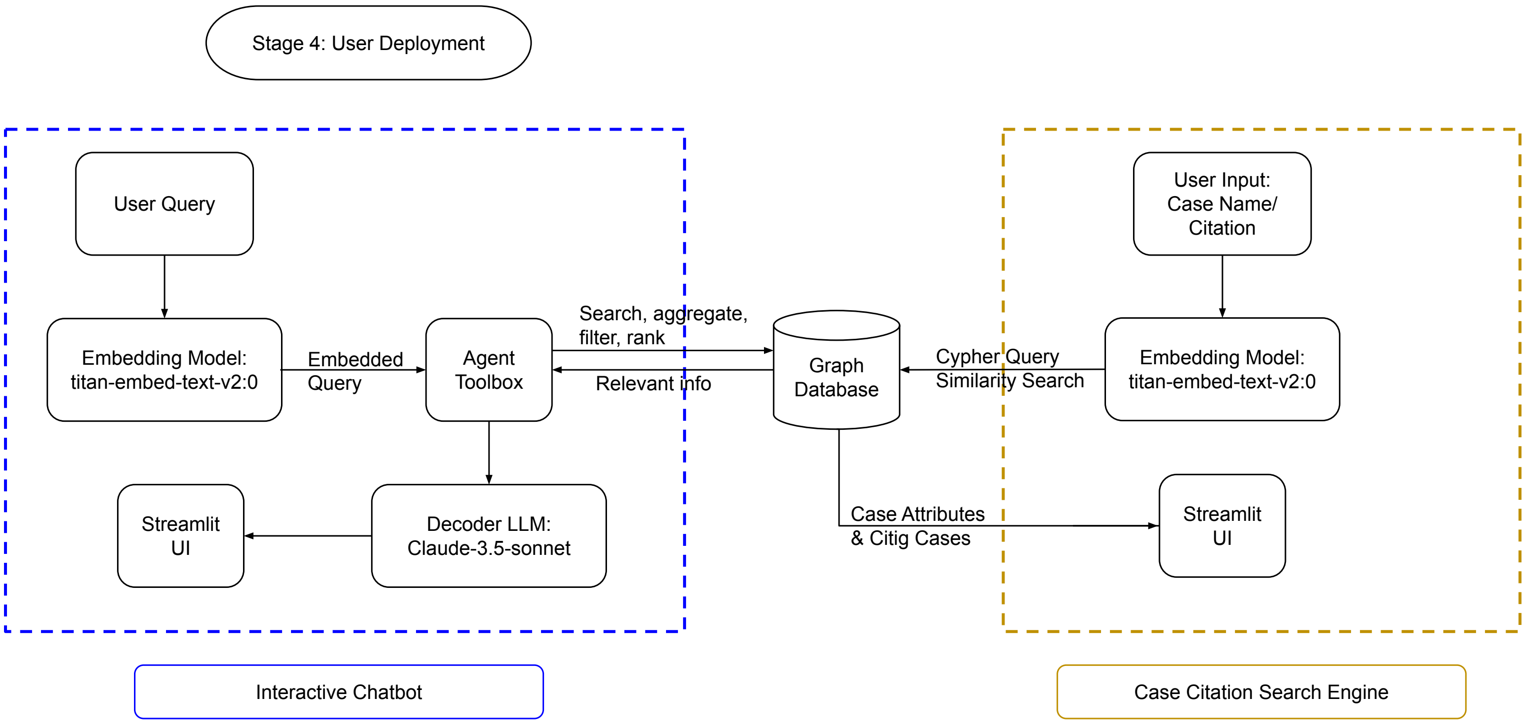
Stage 2: Develop Classifier for Single Citation for Label with Rationale



Stage 3: Develop Computational Algorithm for Case Classification



Stage 4: Deploy to end user with interactive search machine and chatbot design

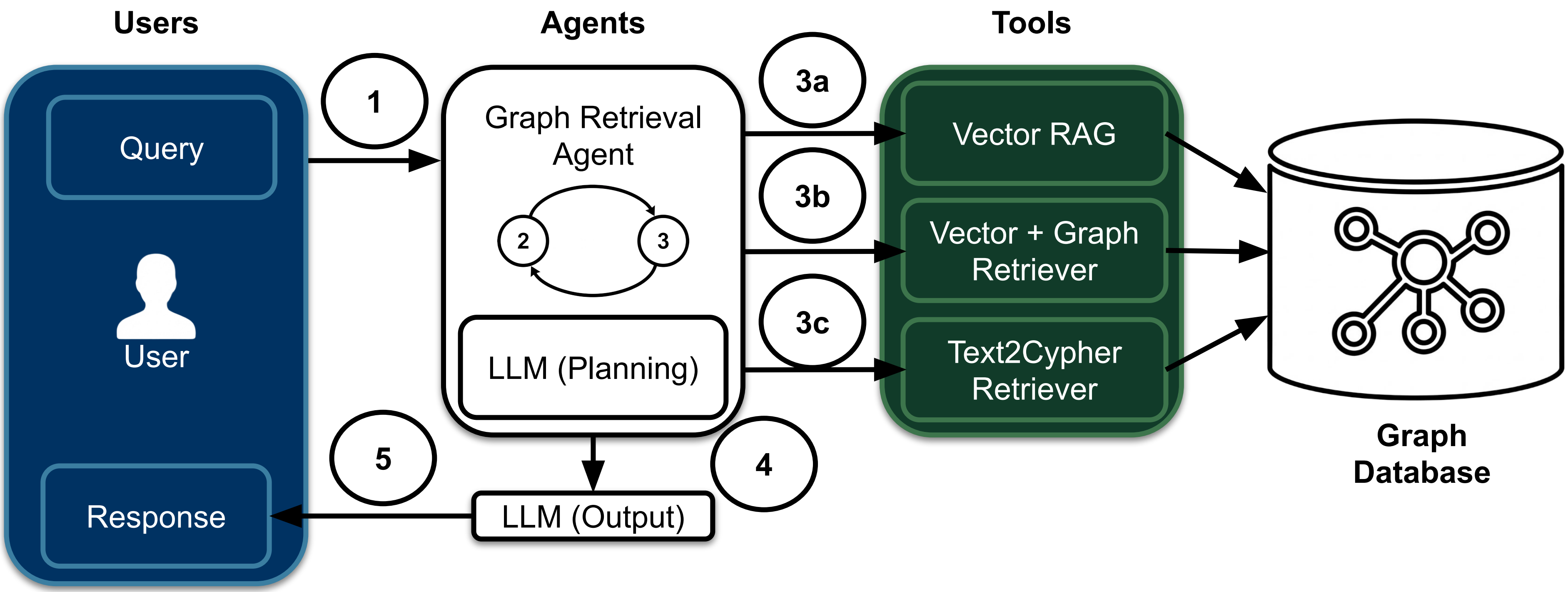


Technical Approaches

Building an Interactive Chatbot

LLM = Claude 3.5 Sonnet
Embedder = Amazon Titan Text Embeddings V2

Agentic GraphRAG Framework



Knowledge Graph

Dataset, EDA, ETL Pipeline, Knowledge Graph

Dataset

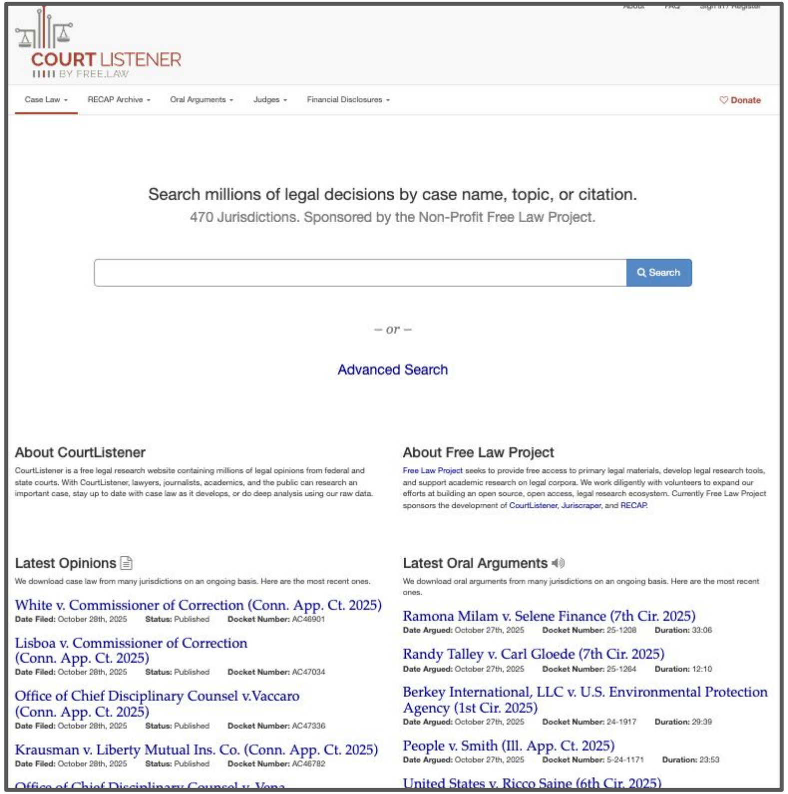
Stage 1: Build a Graph Database for Case Search and Mapping

Source

- Americans with Disabilities Act Handbook (ADAH)
- CourtListener

Case Range: Cases in ADAH and cases citing ADAH cases

COURT LISTENER
BY FREE·LAW

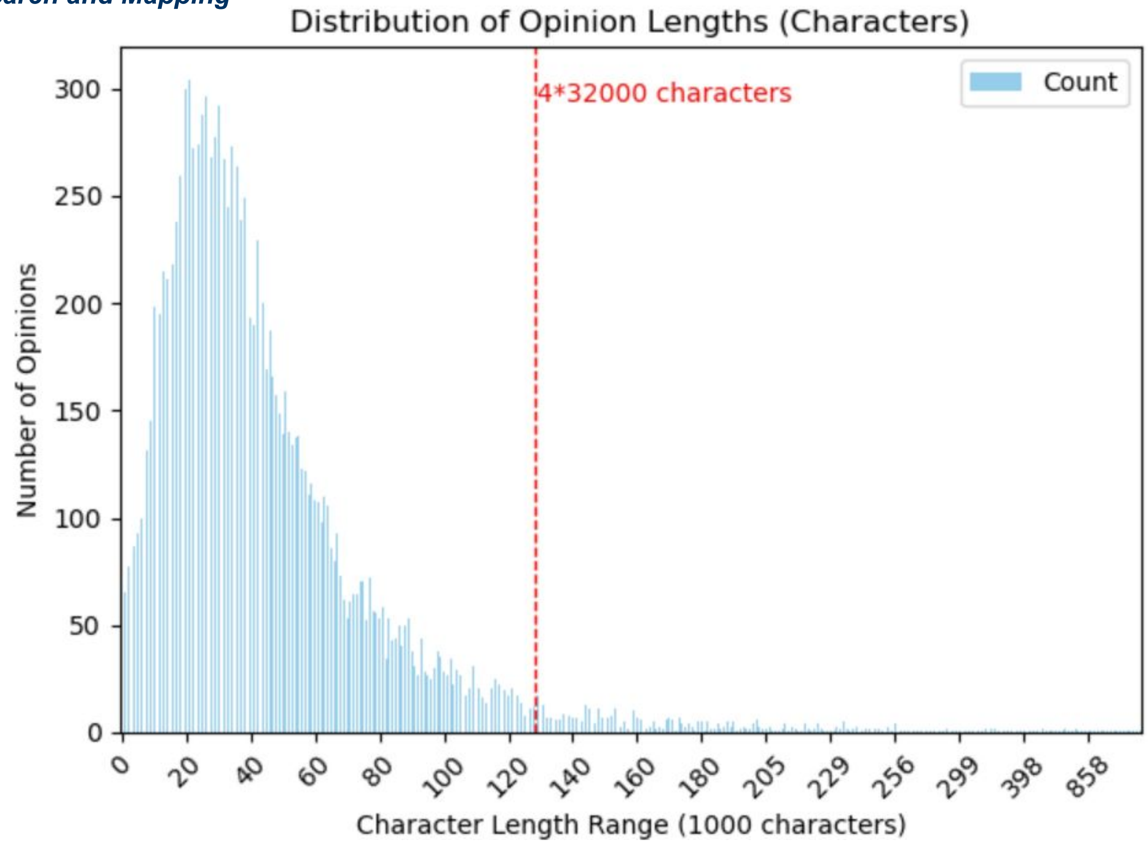


EDA

Stage 1: Build a Graph Database for Case Search and Mapping

Case Length

A model with context length > 32,000 tokens will cover 95% of the cases from total dataset.



Knowledge Graph

ETL Pipeline

Stage 1: Build a Graph Database for Case Search and Mapping

2.6k ADAH Cases

13.2k Citing Cases

combined cases

Clean Dedup

cleaned cases

Add Features

15.4k final dataset (json)

Sample: 3648 cases

Sentence-aware chunking

dataset with chunks (json)

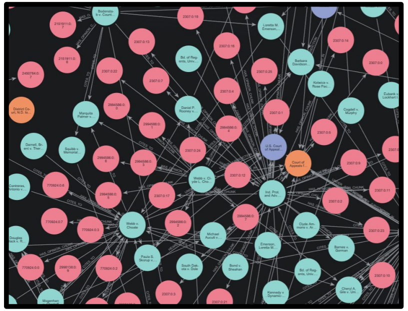
Courts (csv)

Cases (csv)

...

Opinion Chunks (csv)

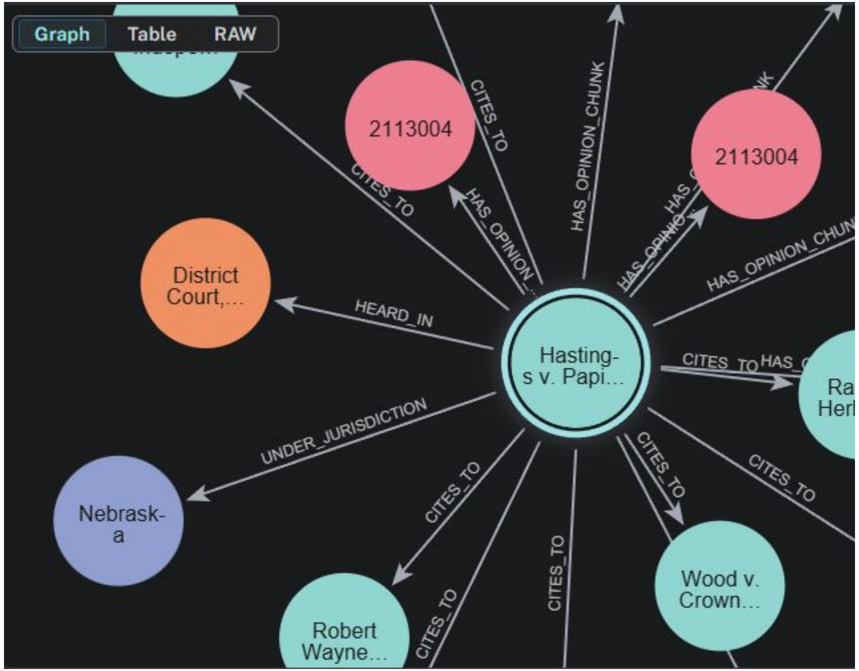
neo4j aura

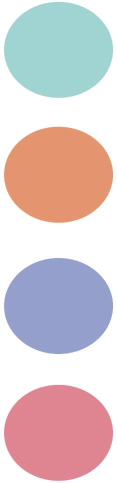


- **Standardize white space, remove control characters**
- **Add court level, jurisdiction**
- **Sentence-aware chunking method**

Knowledge Graph for Case Search and Citation Mapping

Stage 1: Build a Graph Database for Case Search and Mapping





- Cases: 3,648
  - ADAH cases: 410
  - Non-ADAH cases: 3,238
- Courts: 241
- Jurisdictions: 116
- Opinion chunks: 56,347

:CITES_TO Relationships: 5,491

13.4 chunks per cases/
40k characters per chunk

Citation Classifier

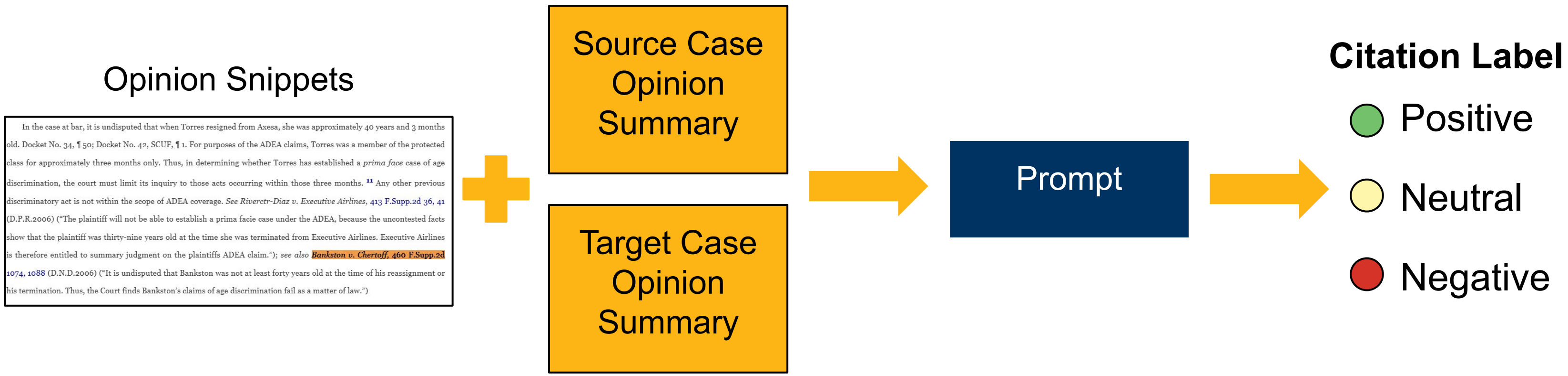
Citation Classifier

Feature Extraction & LLM Prompt Engineering

Stage 2: Develop Classifier for Single Citation for Label with Rationale

Goal: Accurately capture how a citing case treats a cited case.

Approach: Summarize source case and target case opinions. Engineer prompt to classify citation with positive, neutral, and negative labels.



DEFINITIONS:
- **Positive:** The citing court relies on, follows, or agrees with an earlier decision. The cited case serves as authoritative support for the citing court's reasoning or holding.
- **Neutral:** The citing court mentions, describes, or explains an earlier case without expressing approval or disapproval. The reference may provide procedural context, general background, or illustrate a contrasting outcome without evaluating authority.
- **Negative:** The citing court rejects, limits, criticizes, or overrules an earlier decision's reasoning or holding. The cited case is treated as weakened or incorrect authority, indicating that its doctrine should not be relied upon in the same way.

Conducted Experiments from Baseline while Balancing Performance and Tradeoffs

Stage 2: Develop Classifier for Single Citation for Label with Rationale

	Experiments to Tune Classifier	Strength	Weakness
1	Use full opinions rather than summary for LLM prompt	Has full context and helps with reasoning through text	Tends to label most cases as neutral (possible hallucination / overgeneralization)
2	Change classification sequence for LLM - example 1: Positive, Negative, Neutral - example 2: Negative, Positive, Neutral	The first label listed has disproportionate influence. Can shift the model toward identifying more of that label	Causes bias toward whichever label appears first. Reduces reliability across classes.
3	Impose strict rules for LLM to follow for Positive citation classification	Increases the model's sensitivity to positive cues. Helps catch more true positives	Still struggles to separate neutral vs. positive cleanly
4	Add more examples for LLM to learn	Gives clearer reference points for the model	Did not significantly improve performance

Final Citation Classification via 3-Model Ensemble

Stage 2: Develop Classifier for Single Citation for Label with Rationale

Evaluation Set: 36 Instances (17 positive, 10 neutral, 9 negative)

LLMs	Accuracy	Precision
Mistral 7B	36%	57%
Claude 3.5 Sonnet	67%	71%
LLaMA 3 (70B)	61%	56%
Ensemble	67%	70%

Final Selection: Ensemble with “Majority Voting”

- Using all three models and assign each citation a label based on majority agreement across the models.

Key Takeaways for using Ensemble

- ➔ Provides a safeguard by reducing dependence on any single model and its biases.
- ➔ Offers a more conservative, panel-style decision by using three independent “reviewers” and taking the majority vote.
- ➔ Produces a final citation label that reflects broad model agreement.

Ensemble: Illustrating how majority vote works

Stage 2: Develop Classifier for Single Citation for Label with Rationale

Method: Each model picks a citation label, the final label is chosen by the majority vote

Cited Case: *Frazier v. Simmons*

Citing Case: *Acevedo v. City of Philadelphia*

Model	Predicted Label
Mistral 7B	Neutral
Claude 3.5 Sonnet	Neutral
LLaMA 3 (70B)	Positive
Final Ensemble Label	Neutral

Takeaway: Ensemble voting reduces single-model bias and improves label stability.

Note: If all three models have different labels, then global label is Neutral

Comparison to Industry Models

Stage 2: Develop Classifier for Single Citation for Label with Rationale

Model	Accuracy
Google Gemini Pro 3	69%
OpenAI GPT-5	67%
Claude Sonnet 4.5	56%
Final Ensemble Label	**67%**

Takeaway: Ensemble method performs on par with best-in-class performance from models like GPT-5 and Gemini Pro 3

Case Classifier with Computational Algorithm

Feature Engineering for Case Classification

Stage 3: Develop Computational Algorithm for Case Classification

Goal:

- Assign each case a label: **Good**, Bad, **Moderate**, or Unknown.

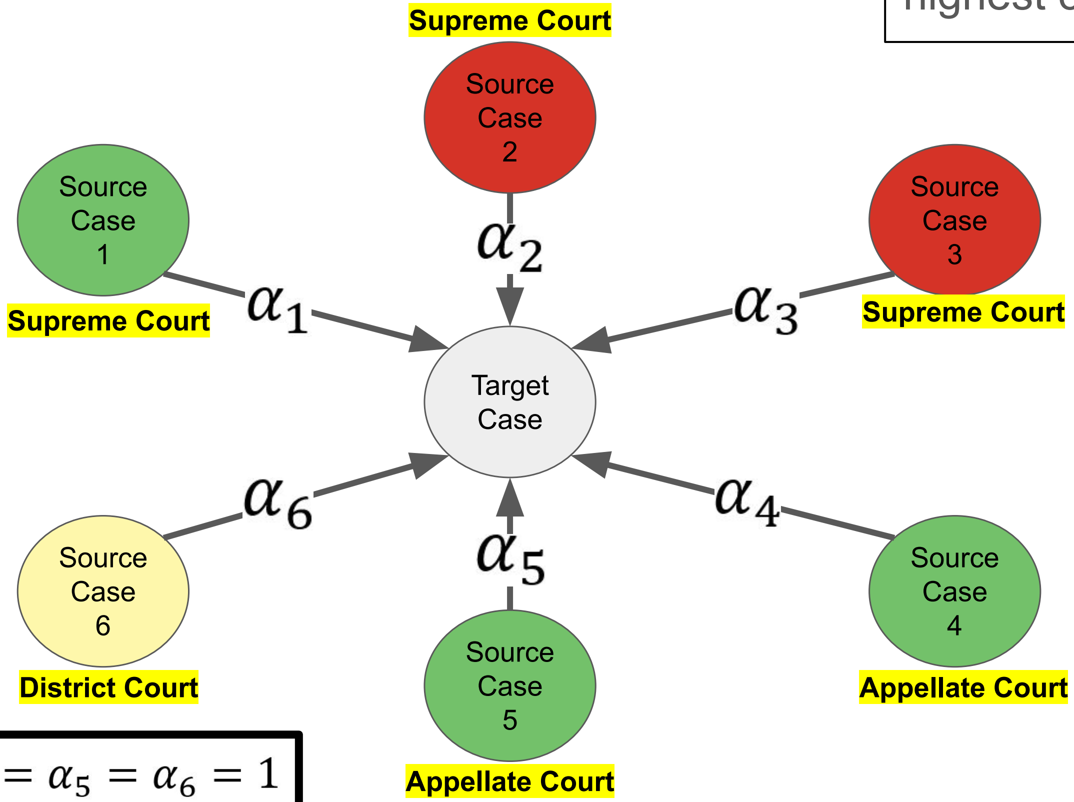
Important features:

- **Court Level (1, 2, 3, 4, 5)**
    1. Federal Supreme and Appellate
    2. Federal Appellate Courts (Court of Appeals)
    3. Federal District/Trial Courts
    4. State Courts
    5. No court available from CourtListener API
- **Citation Classification**
- **Case Decision Date**
- **Jurisdiction**

Case Classification - Example #1

Stage 3: Develop Computational Algorithm for Case Classification

Step 1: Start with the highest court level



$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

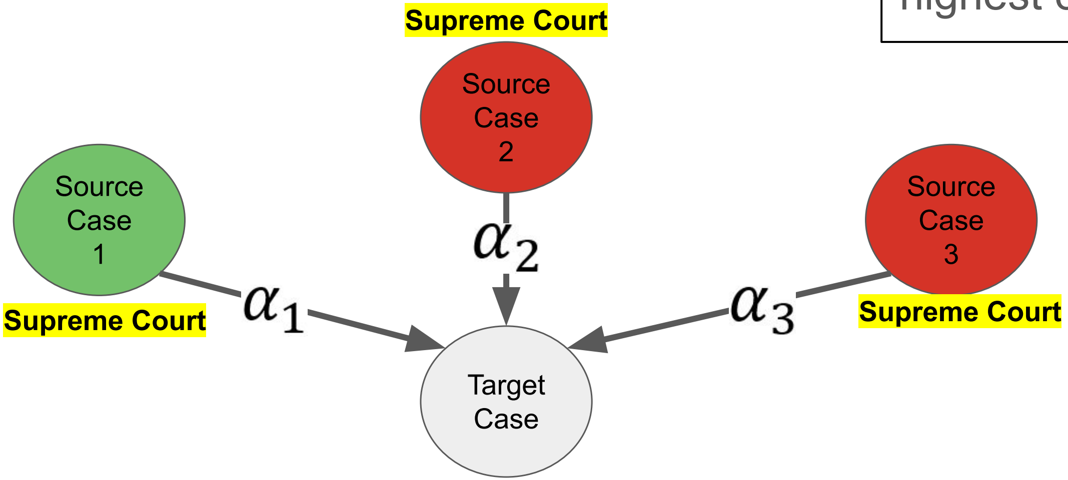
Citation Classification

- Positive
- Neutral
- Negative

Case Classification - Example #1

Stage 3: Develop Computational Algorithm for Case Classification

Step 1: Start with the highest court level



$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

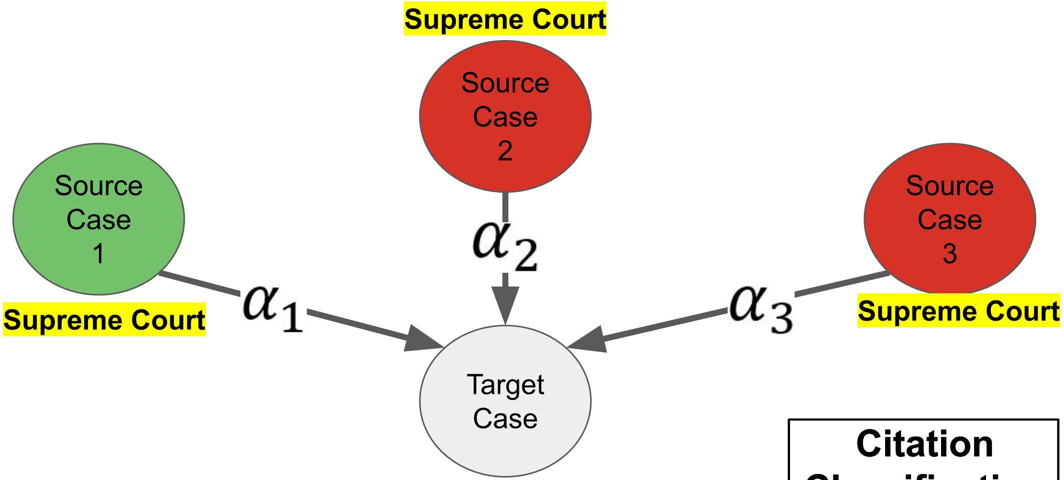
Citation Classification

- Positive
- Neutral
- Negative

Case Classification - Example #1

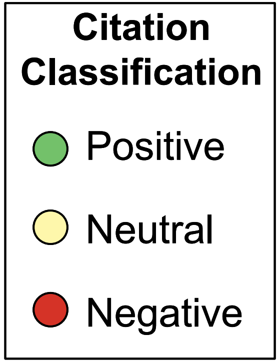
Stage 3: Develop Computational Algorithm for Case Classification

Step 2: Compute Label Proportions



$$p_{Pos} = \frac{1}{3} = 0.33$$

$$p_{Neg} = \frac{2}{3} = 0.67$$



$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

Case Classification - Example #1

Stage 3: Develop Computational Algorithm for Case Classification

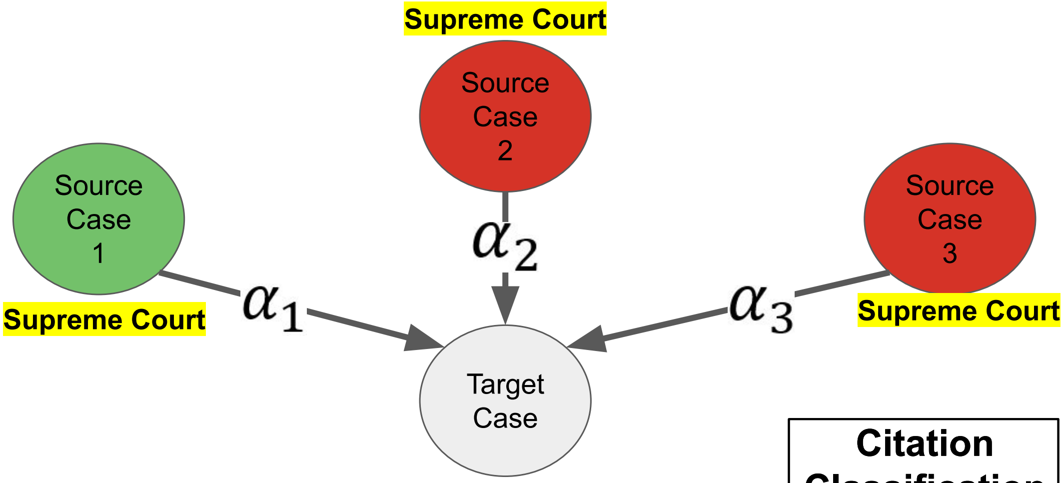
Step 3: Decide the dominant treatment

$$p_{pos} \geq threshold_{pos} = 0.50$$

$$\boxed{p_{Neg} \geq threshold_{Neg} = 0.50}$$

$$p_{Neu} \geq threshold_{Neu} = 0.50$$

$$p_{Unk} \geq threshold_{Unk} = 0.50$$



Citation Classification

- Positive
- Neutral
- Negative

$$p_{Pos} = \frac{1}{3} = 0.33$$

$$\boxed{p_{Neg} = \frac{2}{3} = 0.67}$$

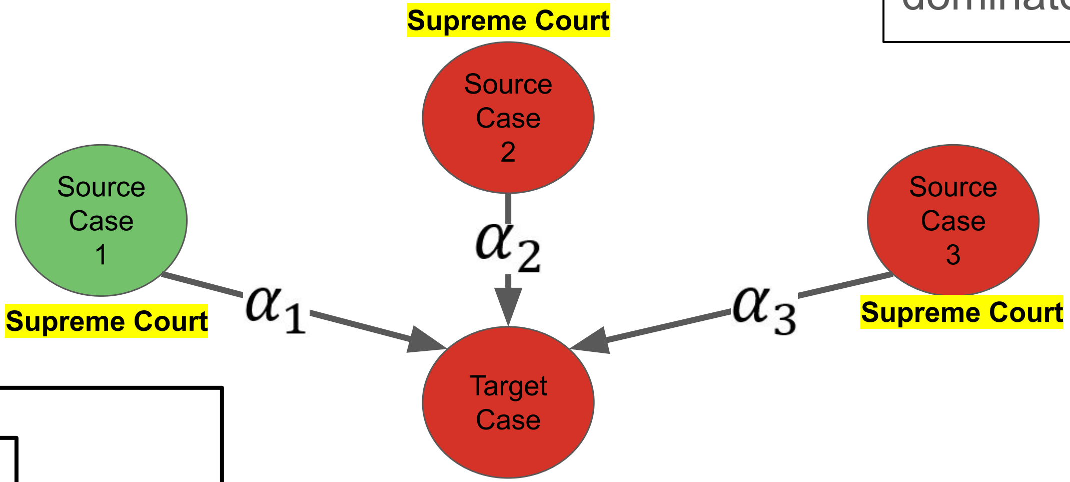
$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

Case Classification - Example #1

Stage 3: Develop Computational Algorithm for Case Classification

Step 4: If treatment dominates, **label case**

Negative Treatment Dominates



Map Treatment to Case Label:
Positive → **Good**
Negative → **Bad**
Neutral → **Moderate**
Unknown → **Unknown**

Target Case is Labeled: **“Bad”**

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

Citation Classification

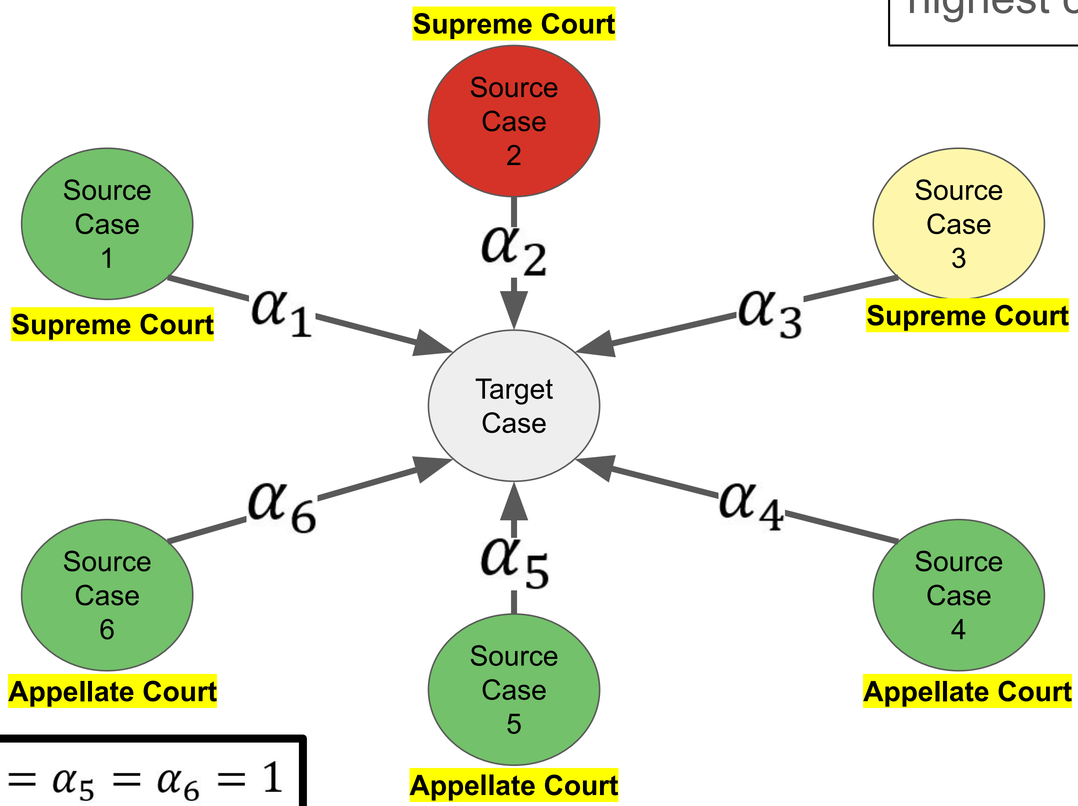
- Positive
- Neutral
- Negative

Case Classification - Example #2

Stage 3: Develop Computational Algorithm for Case Classification

Step 1: Start with the highest court level

What if?



$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

Citation Classification

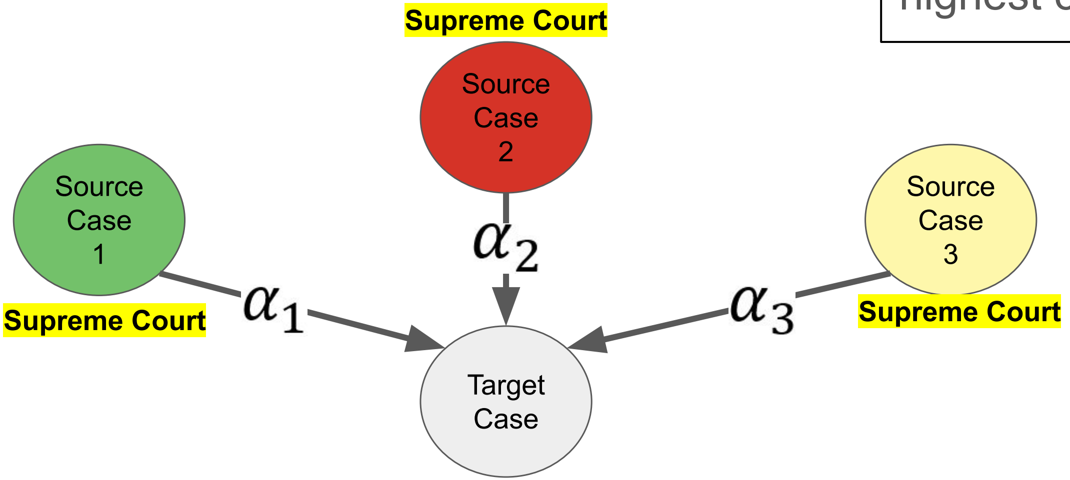
- Positive
- Neutral
- Negative

Case Classification - Example #2

Stage 3: Develop Computational Algorithm for Case Classification

Step 1: Start with the highest court level

What if?



$$p_{Pos} = \frac{1}{3} = 0.33$$

$$p_{Neg} = \frac{1}{3} = 0.33$$

$$p_{Neu} = \frac{1}{3} = 0.33$$

$$p_{pos} \geq threshold_{pos} = 0.50$$

$$p_{Neg} \geq threshold_{Neg} = 0.50$$

$$p_{Neu} \geq threshold_{Neu} = 0.50$$

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

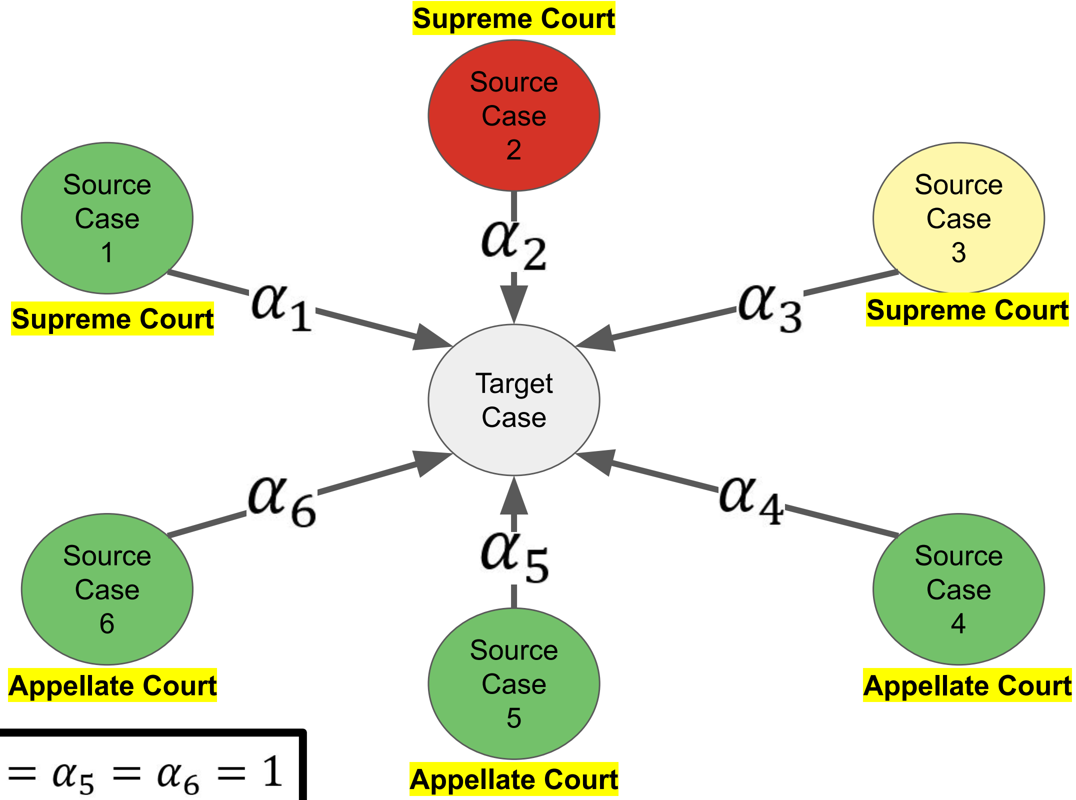
Citation Classification

- Positive
- Neutral
- Negative

Case Classification - Example #2

Stage 3: Develop Computational Algorithm for Case Classification

No dominant treatment at higher court, then analyse lower courts



$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

Citation Classification

- Positive
- Neutral
- Negative

Case Classification - Example #2

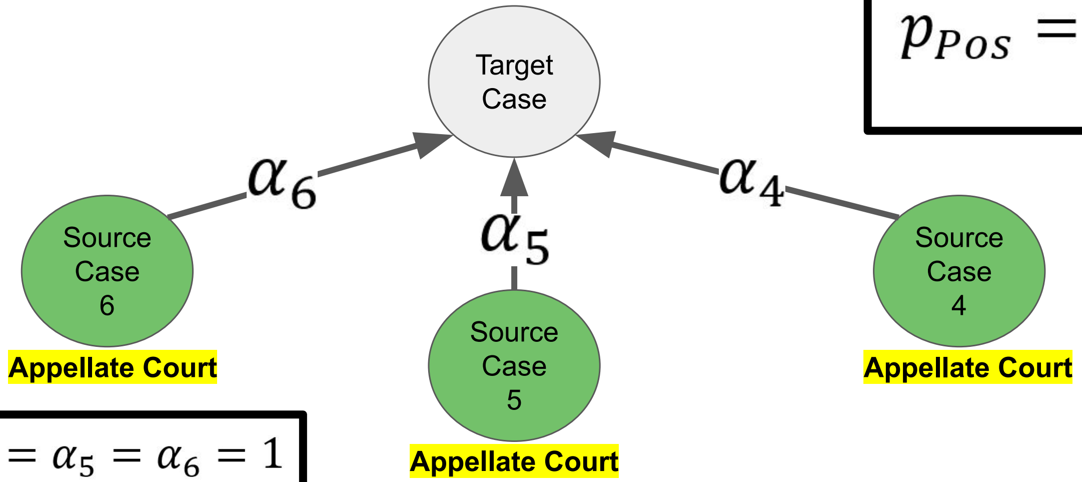
Stage 3: Develop Computational Algorithm for Case Classification

Step 2: Compute Label Proportions
Step 3: Decide the dominant treatment

Positive
Treatment
Dominates

$$p_{pos} \geq threshold_{pos} = 0.50$$

$$p_{Pos} = \frac{3}{3} = 1.0$$



$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

Citation Classification

- Positive
- Neutral
- Negative

Case Classifier with Computational Algorithm

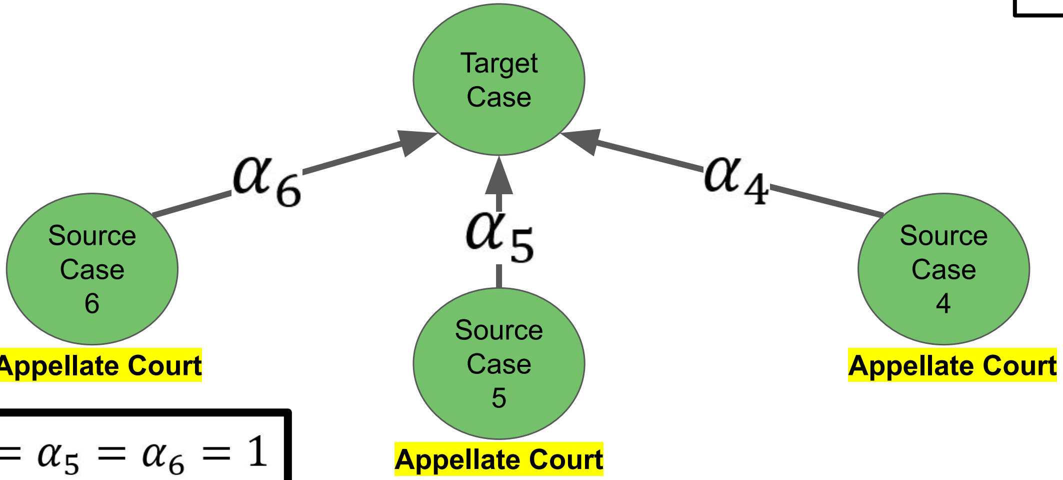
Case Classification - Example #2

Stage 3: Develop Computational Algorithm for Case Classification

Step 4: If treatment dominates, **label case**

Target Case is Labeled: **"Good"**

Map Treatment to Case Label:
Positive → **Good**
Negative → **Bad**
Neutral → **Moderate**
Unknown → **Unknown**



Citation Classification

Positive

Neutral

Negative

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \alpha_6 = 1$$

Case Labeling - Scoring Mechanism with Case Recency and Jurisdiction Factor

Stage 3: Develop Computational Algorithm for Case Classification

$\boldsymbol{\alpha}$(e) = Citation Score

Where $\boldsymbol{\alpha}$**(e)** ∈ [1,MAX WEIGHT]

MAX WEIGHT:
How much more recent citations should count than older ones.

$$\alpha(e) = 1 + (\text{MAX WEIGHT} - 1) r_e$$

A Function of Case **Decision Date (Recency)** r_e

$$\alpha(e) = \alpha_{\text{base}}(e) + J_i$$

J_i → **Jurisdiction Factor**

Giving users the control of the computational algorithm

Stage 3: Develop Computational Algorithm for Case Classification

Control the Signal Strength

Adjust treatment share thresholds

$$p_{\text{label}} \geq \text{threshold}_{\text{label}}$$

Give more Weight to Recent Law

Boost recent citations or narrow the time window

Focus on Jurisdictions that Matter

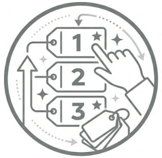
Add jurisdiction weights so key courts drive the label more

Choose your Court Strategy

"Highest Court Only" or "Walk Down" Strategy

Resolve close calls your way

Set a label priority order so the system breaks ties



Conclusion

Innovations, Challenges, Roadmap

Innovations

- **Case semantics via Knowledge Graph** layer with case and citation context
- **Case snippet and opinion summaries** for rich context and noise reduction
- **3-LLM model ensemble** for reliability and algorithm transparency
- **Agent and Graph-RAG implementation** for advance conversational chatbot
- **Interactive user friendly solution** with user control prioritized

Conclusion

Roadmap

Use **graph analytics to deepen case evaluation** (centrality, community detection, influence)

Parallelize the pipeline to **reduce runtime** when summarizing and classifying citations

Fine-tune an LLM on a larger labeled dataset for **domain-specific citation evaluation**

Deploy in a **production setting**

Innovate legal research with AI-powered citation analysis

Thank You!

Questions?