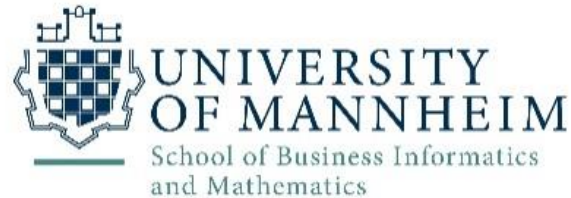


Agent-driven Code Recommendation Service (ACRS) using LangChain and LASSO

Team Project

**Chair of Software Engineering (Prof. Atkinson) &
Chair of Web-based Systems (Prof. Bizer)**

Fall Semester 2025



An Agent-driven Code Recommendation Service

- Summary
 - Develop an AI-driven, agent-based service that provides personalized code recommendations based on analysis of large numbers of software implementations, mimicking the functionality of LASSO's code analysis pipelines
- Project involves
 - **Agents:** Use autonomous software agents to drive the code search process
 - **Generative AI:** Use of LLMs to enable conversations and perform tasks
 - **Code Recommendation:** The primary focus is on searching for / or generating *relevant* code snippets or implementations
 - **Services:** The provision of a service-oriented agent framework that offers code recommendation capabilities
- Technology Components
 - LASSO platform: automate the gathering and analysis (including static and dynamic analysis) of large numbers of harvested software implementations (<https://github.com/SoftwareObservatorium/lasso>)
 - LLMs for Software Engineering tasks
 - LangChain: AI toolkit for integration and agents (<https://www.langchain.com/langchain>)



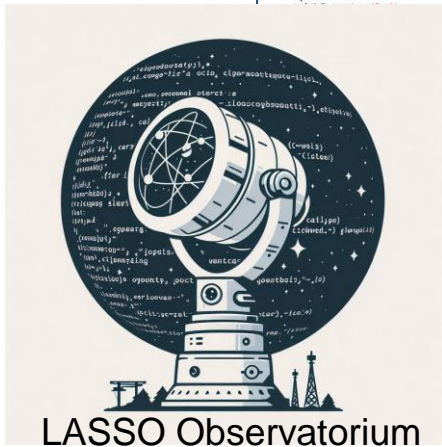
LASSO – Code Recommendation Service

LSL Analysis Pipelines

```

// test from given data source
// define some helper functions
studyName := "org.mortbay.util.ArrayQueue"
in := read("data")
actionName := "enqueue"
abstractClass := "Queue"
// extend some existing java class (dummy)
queryForClasses := "java.lang.Object"
// define the pipeline
profile := {
  name := "enqueue"
  environment := "java"
  language := "java"
}
// populate and create the
action := {
  name := "enqueue"
  type := "enqueue"
  sequence := [
    {
      name := "enqueue"
      row := "enqueue"
      col := "enqueue"
    }
  ]
  // other tests
}
// dependencies
dependencies := {
  includeDependencies := "Stack" // select implementation from former action
  profile := "enqueue"
}
// measure fit of
actionName := "enqueueScore", type := "Fitout"
dependencies := "enqueue"
    
```

	A	B	C	D
1	enqueue	create	Queue	
2		enqueue	A1	"hello"
3	"hello"	enqueue	A1	"world"
4	2	size	A1	
5	"hello"	peek	A1	
6	1	size	A1	



1) Query



Interface Signature

Specify your interface in LQL

Clear

```

1 Queue {
2   enqueue(Object) -> Object
3   dequeue() -> Object
4   peek() -> Object
5   size() -> int
6 }
    
```

Sequence Sheets

Specify one or more Sequence Sheets

Sheet 1

	A	B	C	D	E
1		create	Queue		
2		enqueue	A1	"hello"	
3		enqueue	A1	"world"	
4	"hello"	peek	A1		
5	2	size	A1		
6	"hello"	dequeue	A1		
7	1	size	A1		

Add Row Add Column New Sheet Load Export

Settings

Fine-tune parameters

2) Manually pick best match (preferences)

Results for LSL Pipeline

1ba2d2e3-666f-4bb0-8707-68cc4d1d7bb0

<> View LSL Script SRM Database

Items per page: 25 1 - 25 of 475

#1 org.mortbay.util.ArrayQueue ²⁵

org.mortbay.jetty-util:5.6.1.26

Interface Code Responses Operations Stimuli Observations Reports

```

1 ArrayQueue {
2   ArrayQueue()
3   ArrayQueue(int)
4   ArrayQueue(int, int)
5   ArrayQueue(int, int, java.lang object)
6
7   getCapacity() -> int
8   add(java.lang object) -> boolean
9   offer(java.lang object) -> boolean
    
```



LLMs – Code Generation

LLMs (chatbots) trained on massive amounts of open source code



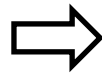
GPT-*



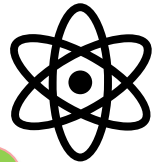
generate a java method that serializes an object to a string using JSON



deepseek



Here's an example Java method, `serializeToJsonString`,
....



GitHub Copilot

... many more



ACRS – Multi-Agent Code Recommendation (1)

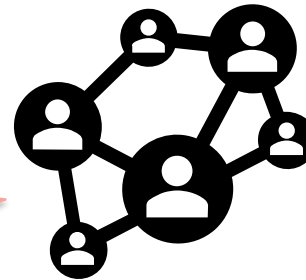


Recommend a Java method that serializes an object to JSON. Pick the best match based on minimal cyclomatic complexity and project popularity.

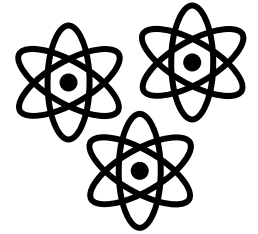
The best match is Java method, toJson,

In addition to cyclomatic complexity, also consider X and Y ...

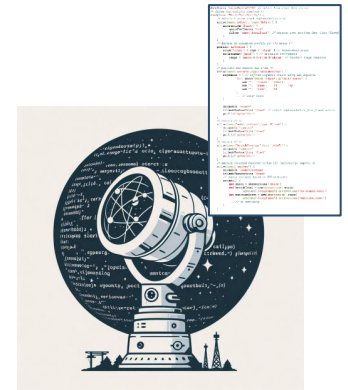
....



Agents



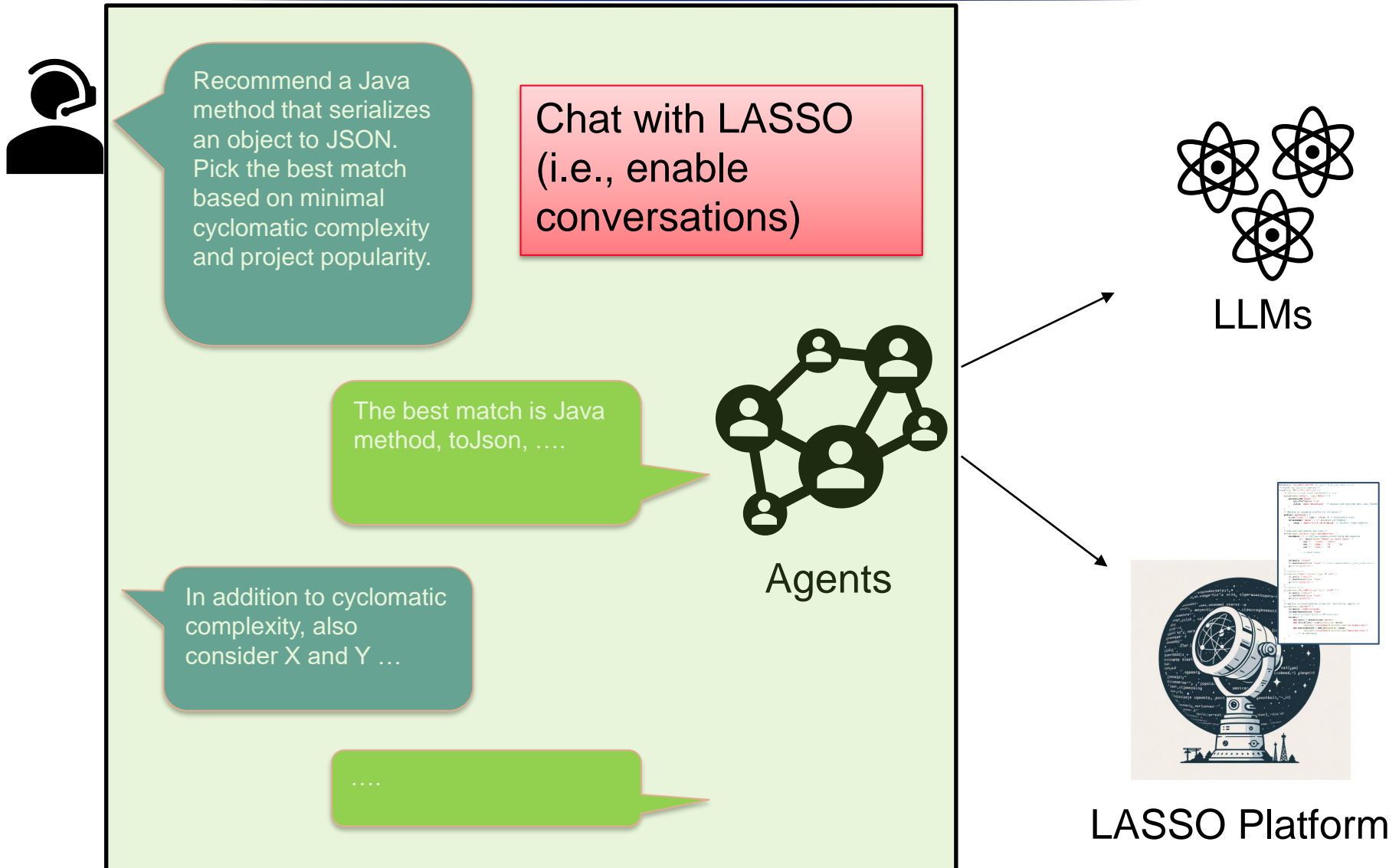
LLMs



LASSO Platform



ACRS – Multi-Agent Code Recommendation (2)



ACRS – Multi-Agent Code Recommendation (3)

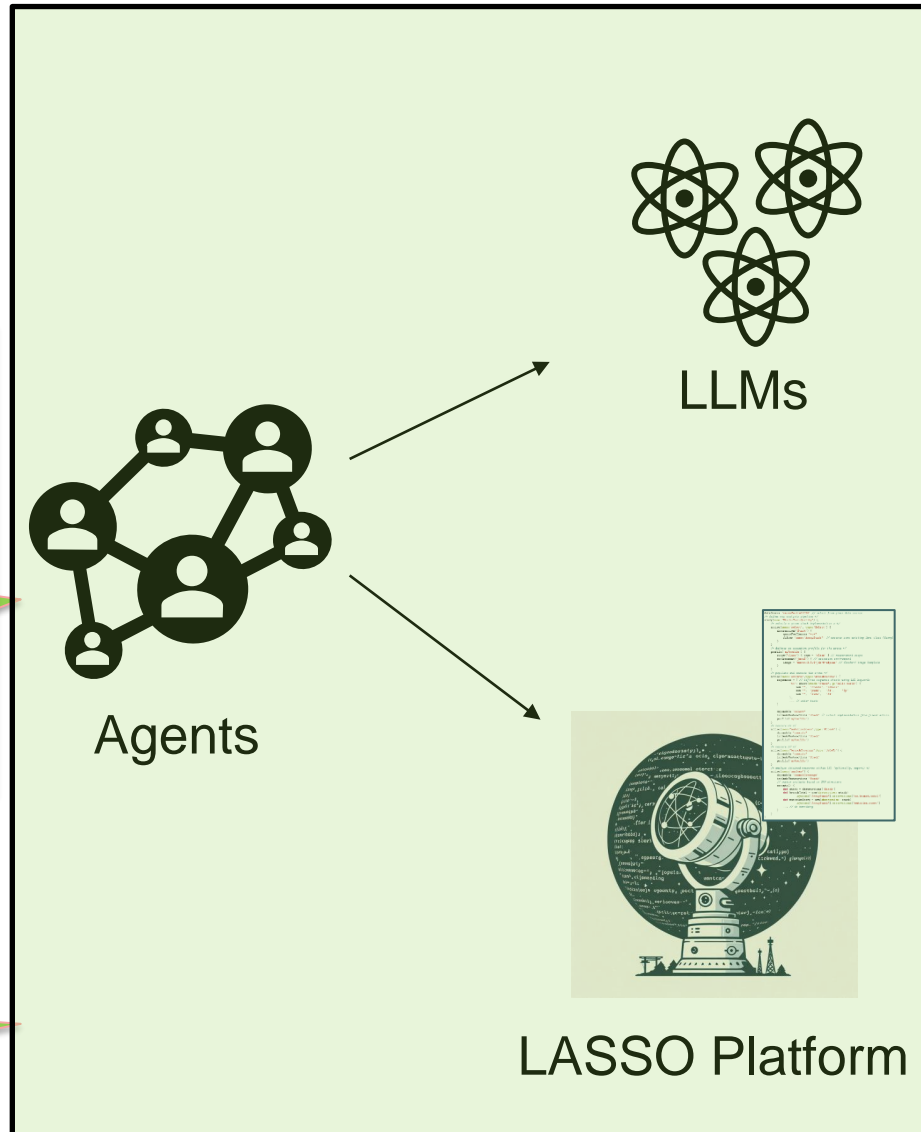


Recommend a Java method that serializes an object to JSON. Pick the best match based on minimal cyclomatic complexity and project popularity

- LangChain: agents + integration
- LLM workflows
- use LASSO's platform services to retrieve "code", "facts" (e.g., metrics) and use analysis "tools" (e.g., execute and analyze code)

complexity, also consider X and Y ...

....



Requirements

- Participants
 - 4-6 students
- Length
 - 6 months
- Prerequisites
 - Python and/or Java programming skills
 - Basic understanding of machine learning
 - Ideally, IE 686 Large Language Models and Agents
- Language
 - English
- Organisation
 - Goals and timetable defined by agreement with the supervisor
- Applicable to MMDS: yes
- Online: By agreement
- Main Supervisor
 - Marcus Kessel
- Technical Supervisor (Agents)
 - Ralph Peeters

