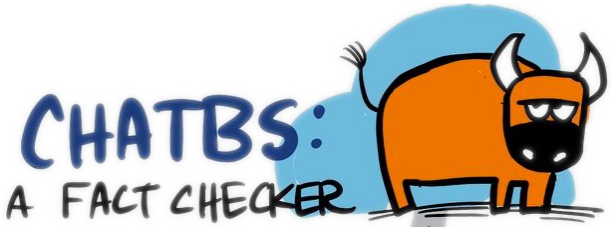
Rensselaer

why not change the world?®



LLM Experimentation through Knowledge Graphs: Towards Improved Management, Repeatability, and Verification

John S. Erickson, Henrique Santos, Vládia Pinheirob, Jamie P. McCusker and Deborah L. McGuinness
with **Nipun Pathirage and Devin De Silva**

February 2025

The Tetherless World Constellation
Rensselaer Polytechnic Institute

Tetherless World Constellation

ScienceDirect

View PDF Download full issue

Outline

Abstract

Keywords

1. Introduction

2. Related works

3. An architecture for redesigning the landscape ...

4. Exploring knowledge graphs: Limitations, rese...

5. Conclusion and future directions

CRediT authorship contribution statement

Declaration of competing interest

Acknowledgments

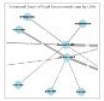
Data availability

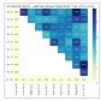
References

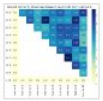
Show full outline

Figures (4)









ELSEVIER

Journal of Web Semantics

Volume 85, May 2025, 100853



LLM experimentation through knowledge graphs: Towards improved management, repeatability, and verification

John S. Erickson [a], Henrique Santos [a], Vládia Pinheiro [b a], Jamie P. McCusker [a], Deborah L. McGuinness [a]

Show more

Add to Mendeley Share Cite

https://doi.org/10.1016/j.websem.2024.100853

Get rights and content

Under a Creative Commons license

open access

Abstract

Generative large language models (LLMs) have transformed AI by enabling rapid, human-like text generation, but they face challenges, including managing inaccurate information generation. Strategies such as prompt engineering, Retrieval-Augmented Generation (RAG), and incorporating domain-specific Knowledge Graphs (KGs) aim to address their issues. However, challenges remain in achieving the desired levels of management, repeatability, and verification of experiments, especially for developers using closed-access LLMs via web APIs, complicating integration with external tools. To tackle this, we are exploring a software architecture to enhance LLM workflows by prioritizing flexibility and traceability while promoting more accurate and explainable outputs. We describe

Agenda

- How we got here: ChatBS Overview & Evolution
- The JoWS Paper: Experimenting with KG-based validation
- The ChatBS-NexGen Project

ChatBS: A Context-aware LLM Exploratory Sandbox

ChatBS version: 2024-03-05 10:52:00 (gpt-4-0613)

https://inciteprojects.idea.rpi.edu/chatbs/app/chatbs/

ChatBS, the Context-aware LLM Exploratory Sandbox uses the **OpenAI Completion API** service to answer questions. Each sentence in a ChatBS result is automatically linked to a Google query to facilate fact-checking. ChatBS can then use the OpenAI API to construct an entity/relation graph of these results in the form **['entity1', 'relationship', 'entity2']**. ChatBS then uses entity linking to look up entities and relationships against Wikidata, constructing a JSON-LD graph as it proceeds.

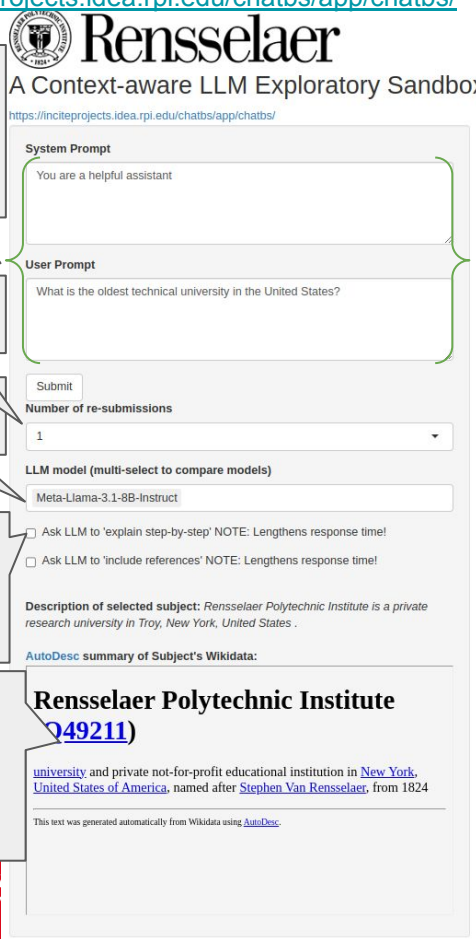
Submit questions to OpenAI Completion API: Custom "System" and "User" prompts

Up to 10 prompt re-submissions

Choice of LLM models (7 total)

Option to append user prompts for "chain-of-thought " reasoning evaluation

Embedded Wikidata entity summary (using **AutoDesc API**) of selected entity…to aid in human fact checking!



Up to 10 unique, independent responses to user's prompt

Each sentence in results is automatically linked to a **Google query**, proving a quick "sanity-check"

Downloadable **JSON results** file includes complete prompts

RDF generation is optional, to facilitate question answering

Table of **RDF triples**, relevant to OpenAI's answers (assertions)

RDF generated by LLM-based **ChatBS-REST** entity linking service

Highlighting when Subject or Object entities are **bnodes** (ie when entities named by ChatGPT are not found in Wikidata)



Downloadable JSON-LD of complete results graph

Simple Question answering & RDF generation with ChatBS

Rensselaer

A Context-aware LLM Exploratory S

https://inciteprojects.idea.rpi.edu/chatbs/app/chatbs/

System Prompt

You are a helpful assistant

User Prompt

What is the oldest technical university in the United States?

Submit

Number of re-submissions

1

LLM model (multi-select to compare models)

Meta-Llama-3.1-8B-Instruct

☐ Ask LLM to 'explain step-by-step' NOTE: Lengthens response time!

☐ Ask LLM to 'include references' NOTE: Lengthens response time!

Description of selected subject: *Rensselaer Polytechnic Institute is a private research university in Troy, New York, United States .*

AutoDesc **summary of Subject's Wikidata:**

Rensselaer Polytechnic Institute (Q49211)

university and private not-for-profit educational institution in New York, United States of America, named after Stephen Van Rensselaer, from 1824

This text was generated automatically from Wikidata using AutoDesc.

ChatBS, the Context-aware LLM Exploratory Sandbox uses the OpenAI Completion API service (GPT and Llama family models) to answer questions. Each sentence in a ChatBS result is automatically linked to a Google query to facilate fact-checking. If requested, ChatBS can then use the OpenAI API to construct an entity/relation graph of these results in the form **['entity1', 'relationship', 'entity2'].** ChatBS then uses entity linking to look up both the entities and relationships against Wikidata entities and properties, constructing a JSON-LD graph as is proceeds.

For each re-submission of the user's prompt we display the LLM response and the resulting RDF (if requested)

LLM Results:

Based on: Meta-Llama-3.1-8B-Instruct

Click on any sentence in results to 'fact-check' with Google (opens new window)

Result 1 : [Meta-Llama-3.1-8B-Instruct] The oldest technical university in the United States is Rensselaer Polytechnic Institute (RPI), which was founded in 1824 in Troy, New York. It was established to focus on the "practical application of science and technology" and has been a leading institution in engineering, science, and technology education for nearly two centuries.

Download complete results as JSON file

Summary of relationships (RDF) found for these results:

Click on a row for more details about that row's subject entity. Objects highlighted in gray are bnodes (blank/unlabelled nodes) in the graph.

☐ Disable relationship discovery and entity linking?

Show 10 entries Search:

	Subject	Relationship	Object	Occurrences
1	Rensselaer Polytechnic Institute	field of work	technology	1
2	Rensselaer Polytechnic Institute	field of work	engineering	1
3	Rensselaer Polytechnic Institute	field of work	science	1
4	Rensselaer Polytechnic Institute	founded by	1824	1
5	Rensselaer Polytechnic Institute	location	Troy	1
6	Rensselaer Polytechnic Institute	instance of	institute of technology	1
7	Rensselaer Polytechnic Institute	instance of	institute of technology	1
8	Rensselaer Polytechnic Institute	country of origin	United States of America	1
9	Rensselaer Polytechnic Institute	on focus list of Wikimedia project	practical application of science and technology	1
10	Rensselaer Polytechnic Institute	inception	1824	1

Showing 1 to 10 of 10 entries Previous 1 Next

Download complete results graph as RDF (JSON-LD)

Demoing "Chain-of-Thought" with ChatBS

Rensselaer

A Context-aware LLM Exploratory Sandbox

https://inciteprojects.idea.rpi.edu/chatbs/app/chatbs/

System Prompt

You are a helpful assistant

User Prompt

What is the oldest technical university in the United States?

Submit

Number of re-submissions

1

LLM model (multi-select to compare models)

Meta-Llama-3.1-8B-Instruct

☑ Ask LLM to 'explain step-by-step' NOTE: Lengthens response time!

☐ Ask LLM to 'include references' NOTE: Lengthens response time!

Description of selected subject: *Rensselaer Polytechnic Institute is a private research university in Troy, New York, United States .*

AutoDesc summary of Subject's Wikidata:

Option to append user prompts for "chain-of-thought " reasoning evaluation

ChatBS, the Context-aware LLM Exploratory Sandbox uses the OpenAI Completion API service (GPT and Llama family models) to answer questions. Each sentence in a ChatBS result is automatically linked to a Google query to facilate fact-checking. If requested, ChatBS can then use the OpenAI API to construct an entity/relation graph of these results in the form **['entity1', 'relationship', 'entity2'].** ChatBS then uses entity linking to look up both the entities and relationships against Wikidata entities and properties, constructing a JSON-LD graph as is proceeds.

For each re-submission of the user's prompt we display the LLM response and the resulting RDF (if requested)

LLM Results:

Based on: Meta-Llama-3.1-8B-Instruct

Click on any sentence in results to 'fact-check' with Google (opens new window)

Result 1 : [Meta-Llama-3.1-8B-Instruct] To find the oldest technical university in the United States, I'll follow these steps: 1. Research historical technical universities in the United States. 2. Identify the dates of establishment for each university. 3. Determine which university was established first. Let's start with step 1: Historical technical universities in the United States include: * Rensselaer Polytechnic Institute (RPI) * Massachusetts Institute of Technology (MIT) * University of Pennsylvania * Worcester Polytechnic Institute (WPI) * Cornell University These universities have a long history of technical education and research. Step 2: Identify the dates of establishment for each university: * Rensselaer Polytechnic Institute (RPI): 1824 * Massachusetts Institute of Technology (MIT): 1861 * University of Pennsylvania: 1740 (note: not strictly a technical university, but offers technical programs) * Worcester Polytechnic Institute (WPI): 1865 * Cornell University: 1865 Step 3: Determine which university was established first: Based on the dates of establishment, Rensselaer Polytechnic Institute (RPI) was established in 1824, making it the oldest technical university in the United States. Therefore, the oldest technical university in the United States is Rensselaer Polytechnic Institute (RPI), established in 1824.

Download complete results as JSON file

Summary of relationships (RDF) found for these results:

Click on a row for more details about that row's subject entity. Objects highlig

☑ Disable relationship discovery and entity linking?

Download complete results graph as RDF (JSON-LD)

[Meta-Llama-3.1-8B-Instruct] To find the oldest technical university in the United States, I'll follow these steps:

1. Research historical technical universities in the United States.
2. Identify the dates of establishment for each university.
3. Determine which university was established first.

Let's start with step 1: Historical technical universities in the United States include:
* Rensselaer Polytechnic Institute (RPI)
* Massachusetts Institute of Technology (MIT)
* University of Pennsylvania * Worcester Polytechnic Institute (WPI)
* Cornell University These universities have a long history of technical education and research.

Step 2: Identify the dates of establishment for each university:
* Rensselaer Polytechnic Institute (RPI): 1824 * Massachusetts Institute of Technology (MIT): 1861
* University of Pennsylvania: 1740 (note: not strictly a technical university, but offers technical programs)
* Worcester Polytechnic Institute (WPI): 1865
* Cornell University: 1865

Step 3: Determine which university was established first: Based on the dates of establishment, Rensselaer Polytechnic Institute (RPI) was established in 1824, making it the oldest technical university in the United States. Therefore, the oldest technical university in the United States is Rensselaer Polytechnic Institute (RPI), established in 1824.

Multiple retries with ChatBS, showing response variations

Rensselaer

A Context-aware LLM Exploratory Sandbox

https://inciteprojects.idea.rpi.edu/chatbs/app/chatbs/

System Prompt

You are a helpful assistant

User Prompt

What is the oldest technical university in the United States?

Up to 10 prompt re-submissions

Submit

Number of re-submissions

5

LLM model (multi-select to compare models)

Meta-Llama-3.1-8B-Instruct

☐ Ask LLM to 'explain step-by-step' NOTE: Lengthens response time!

☐ Ask LLM to 'include references' NOTE: Lengthens response time!

Description of selected subject: *Rensselaer Polytechnic Institute is a private research university in Troy, New York, United States .*

ChatBS, the Context-aware LLM Exploratory Sandbox uses the OpenAI Completion API service (GPT and Llama family models) to answer questions. Each sentence in a ChatBS result is automatically linked to a Google query to facilate fact-checking. If requested, ChatBS can then use the OpenAI API to construct an entity/relation graph of these results in the form **['entity1', 'relationship', 'entity2'].** ChatBS then uses entity linking to look up both the entities and relationships against Wikidata entities and properties, constructing a JSON-LD graph as is proceeds.

For each re-submission of the user's prompt we display the LLM response and the resulting RDF (if requested)

Note the variation in responses (Llama 3.1 model)

LLM Results:

Based on: Meta-Llama-3.1-8B-Instruct, Meta-Llama-3.1-8B-Instruct, Meta-Llama-3.1-8B-Instruct, Meta-Llama-3.1-8B-Instruct

Click on any sentence in results to 'fact-check' with Google (opens new window)

Result 1 : [Meta-Llama-3.1-8B-Instruct] The oldest technical university in the United States is Rensselaer Polytechnic Institute (RPI), which was founded in 1824 in Troy, New York. RPI was established to focus on the study of science and technology, and it is often referred to as the "birthplace of technological innovation in the United States." RPI has a rich history of innovation and has produced many notable alumni, including inventors, engineers, and business leaders. Today, RPI is a world-renowned institution and a leader in fields such as engineering, computer science, and data science.

Result 2 : [Meta-Llama-3.1-8B-Instruct] The oldest technical university in the United States is Rensselaer Polytechnic Institute (RPI), which was founded in 1824 in Troy, New York. It was originally founded as the Rensselaer School and offered courses in chemistry, engineering, and other technical fields. Today, RPI is a well-respected institution that offers a wide range of undergraduate and graduate programs in fields such as engineering, computer science, and business.

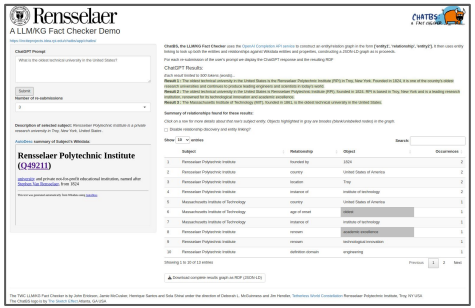
Result 3 : [Meta-Llama-3.1-8B-Instruct] The oldest technical university in the United States is Rensselaer Polytechnic Institute (RPI), which was founded in 1824 in Troy, New York. It was established by Stephen van Rensselaer, a wealthy landowner and politician, with the goal of providing education in the applied sciences and engineering. RPI is often referred to as the "mektep of technology" and has a rich history of innovation and academic excellence.

Result 4 : [Meta-Llama-3.1-8B-Instruct] The oldest technical university in the United States is Rensselaer Polytechnic Institute (RPI), which was founded in 1824 in Troy, New York. It was initially known as Rensselaer School and was established to provide education in the fields of science, technology, engineering, and mathematics (STEM). RPI is consistently ranked among the top technical universities in the world and is known for its rigorous academic programs, cutting-edge research, and innovative spirit.

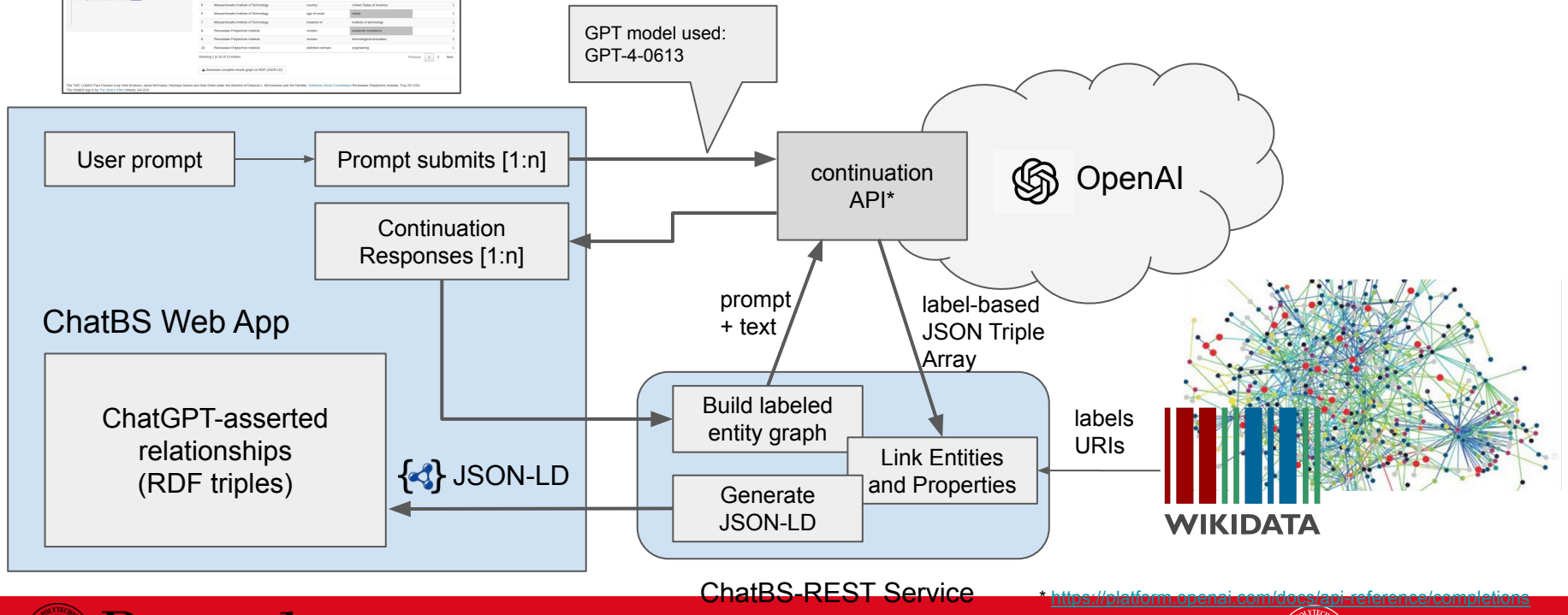
Result 5 : [Meta-Llama-3.1-8B-Instruct] The oldest technical university in the United States is Rensselaer Polytechnic Institute (RPI), which was founded in 1824 in Troy, New York. It was established as the Rensselaer School and was the first independent technological institute in the United States. RPI was originally designed to provide instruction in engineering, chemistry, and other technical fields, and it has a rich history of innovation and entrepreneurship. Today, RPI is a world-renowned institution that offers a wide range of undergraduate and graduate programs in fields such as engineering, computer science, and data science.

Download complete results as JSON file

ChatBS: TWC's LLM/KG FactChecker Demo



The LLM/KG FactChecker uses the OpenAI Completion API service to construct an entity/relation graph in the form **['entity1', 'relationship', 'entity2'].** It then uses **entity linking** to look up both the entities and relationships against Wikidata entities and properties, constructing a JSON-LD graph as is proceeds.



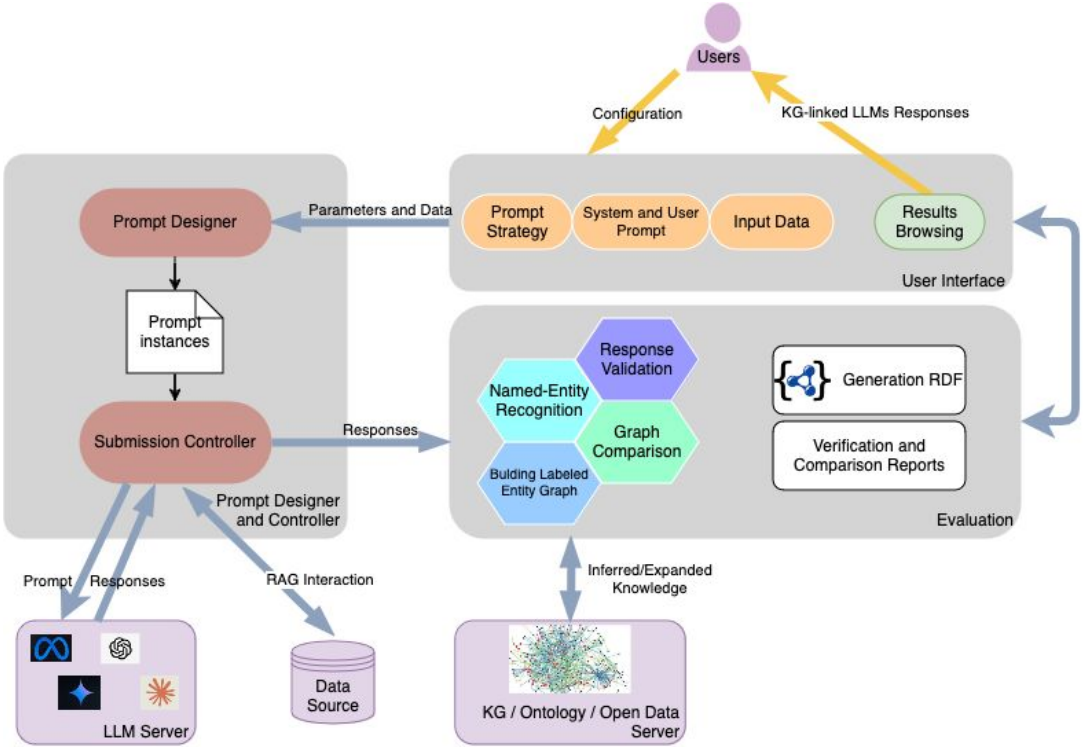
* https://platform.openai.com/docs/api-reference/completions

The JoWS Paper

Based on our *lived experience* with ChatBS, proposed an architecture for improving the management of LLM-based experiments

- Choose prompt strategies: e.g. Few-Shot, Chain of Thought (CoT), Chain of Verification (CoVE), et.al.
- Specifying System and User Prompts with dynamic "slots"
- Select input datasets with values assigned to prompt slots
- Specify the number of prompt resubmissions
- Select from among different (and multiple) LLMs
- Accommodate KG-based evaluation: endpoints, SPARQL templates
- Fine-grained, reproducible experiment logging

ChatBS-NexGen Architecture (from JoWS paper)



The ChatBS-NexGen Experiment (JoWS Paper)

Using ChatBS (manually!), we tested the proposed architecture using a dataset containing data from 100 patients of a "food and nutrition" healthcare program

- A nutrition research group conducts an evaluation of **LLM-generated dietary recommendations** for participants in a health program.
- The goal: **assess the suitability** of the LLM-generated recommendations, taking into account individual factors such as sex, age, gender, weight, health indicators, chronic disease diagnoses, as well as dietary restrictions and preferences.
- Used a live instance of **FoodKG** as our "authoritative KG"

ChatBS-NexGen Nutrition Case Study: Setup

- Prompt Strategy: User selected "Chain of Thought (CoT)" among others strategies - Few-Shot Learning, Chain of Verification (CoVe);
- LLMs: The LLMs to be used in the experiments - for example, GPT-4o mini and Llama 3.1-8B;
- Number of prompt resubmissions: For example, 10 re-submissions to the same LLMs;
- Link for the KG: This link can be an endpoint or a Whyis link. For this example we used the public FoodKG endpoint. [1]
- System prompt:

  You are a health care assistant and you will respond to the user {name}, {age} years old and {gender}, who is a beneficiary of the health care program. Adapt dietary guidelines considering the diagnosis of {diagnosis}, recent data on height: {height}, weight: {weight}, body mass index: {bodyMassIndex}, glycated hemoglobin: glycatedH, blood pressure: {bloodPressure}, hdl: {hdl}, ldl: {ldl}, score of Framingham: {scoreF}, waist circumference: {waistC}. Also consider the food preferences: {preferences}; and allergies: {allergies}.

- User prompt: *I need suggestions for light and energy-boosting foods to eat before my gym workout.*
- Data file with values according to the slots in the System_Prompt and User_prompt above.

2) Prompt Designer and Controller: According to the submitted user information, ChatBS-NexGen executes the Prompt Designer sub-module and instantiates 100 prompts to be submitted to the selected GPT-4o mini and Llama 3 APIs. For example, a prompt instance was generated for a specific patient ANONYMOUS098 - a woman, 53 years old, Diabetic, Vegan, and with no allergies:

You are a health care assistant and you will respond to the user ANONYMOUS098, 53 years old and Female, who is a beneficiary of the health care program. Adapt dietary guidelines considering the diagnosis of DIABETIC, recent data on height: 170.4, weight: 103.7, body mass index: 35.7, glycated hemoglobin: 8.4, blood pressure: 120/80 mmHg, hdl: 58.9, ldl: 109.0, score of Framingham: 12.0, waist circumference: 88cm. Also consider the food preferences: VEGAN; and allergies: NONE.

ChatBS-NexGen Nutrition Case Study: Submission and Results

The **Submission Controller** sub-module submits (and re-submits 10 times) each instantiated prompt to all selected LLMs, receives their responses, and passes them along to the Evaluation step. For example, for the patient **ANONYMOUS098** described above, **Llama 3.1-8B** gave the following recommendation in the first submission:

Result 1 : [MetaLlama3.18BInstruct] Based on your dietary preferences and restrictions, I'd like to suggest some light and energy-boosting foods that are perfect for pre-workout fueling: 1. Banana with almond milk: A classic combo that's easy to digest and provides sustained energy. Choose a ripe banana and almond milk that's unsweetened and fortified with vitamin D. 2. Avocado toast: Whole-grain toast topped with mashed avocado, salt, and pepper is a satisfying and filling option. Look for vegan-friendly bread options. 3. Oatmeal with fruit: Steel-cut oats or rolled oats cooked with plant-based milk and topped with your favorite fruits (such as berries, sliced banana, or diced apples) provide complex carbohydrates for sustained energy. 4. Energy balls: Mix rolled oats, nut butter (such as peanut butter or almond butter), and ***honey*** *or* ***maple syrup*** *to create bite-sized energy balls. These no-bake bites are easy to make and pack. 5. Smoothie bowl: Blend your favorite fruits, spinach, and plant-based milk, then top with granola, nuts, or seeds for added crunch. Choose a bowl that's low in sugar and rich in fiber. 6. Whole-grain crackers with hummus: Spread hummus on whole-grain crackers for a satisfying snack that's high in protein and fiber. 7. Coconut water: This natural electrolyte drink helps replenish fluids and provides a refreshing pick-me-up. 8. Dark chocolate: Enjoy a small piece of dark chocolate (at least 70% cocoa) for a mood boost and energy kick. Just be mindful of portion sizes! 9. Trail mix: A mix of nuts, seeds, and dried fruits is a convenient snack that's easy to devour on-the-go. Look for vegan-friendly options without added sugars. 10. Green juice: Juice fresh greens like kale, spinach, or collard greens with lemon juice and apple cider vinegar for a nutrient-dense drink that's easy to digest. Remember to stay hydrated by drinking plenty of water before, during, and after your workout. Aim to eat your snack 30-60 minutes prior to exercise to allow for digestion and optimal energy levels. Enjoy your workout*

ChatBS-NexGen Nutrition Case Study: KG-checking the LLM

- Recognize the food names mentioned in the Llama-3.1 response
- Link each of the food names into FoodKG and the "WhatToMake" ontology
- Retrieve each food's respective Glycemic Index (GI) -- if available
- Provide information that can help nutrition professionals assess whether LLM recommendations are appropriate for a particular health condition – DIABETES, for example!

Food Recommended	FoodKG.ingredientName	GI
"almond milk"	"almond milk"	30
"apple cider vinegar"	"apple cider vinegar"	5
"apple"	"apple"	
"avocado"	"avocado"	10
"banana"	"banana"	48
"berry"		
"coconut water"	"coconut water"	
"dark chocolate"	"dark chocolate"	23
"dried fruit"	"dried fruit"	
"energy ball"		
"fruit"	"fruit"	
"granola"		
"green juice"		
"honey"	"honey"	60
"hummus"	"hummus"	25
"kale"	"kale"	
"lemon juice"	"lemon juice"	20

Food Recommended	FoodKG.ingredientName	GI
"maple syrup"	"maple syrup"	65
"nut butter"	"nut butter"	
"nut"		
"oatmeal"	"oatmeal"	40
"peanut butter"	"peanut butter"	
"pepper"	"pepper"	
"plant-based milk"		
"rolled oat"	"rolled oat"	
"salt"	"salt"	0
"seed"		
"smoothie bowl"	"smoothie bowl"	
"spinach"	"spinach"	5
"steel-cut oat"		
"trail mix"	"trail mix"	
"water"	"water"	
"whole-grain cracker"		
"whole-grain toast"		

ChatBS-NexGen Nutrition Case Study: Evaluation

- In this experiment with 100 patient data, we submitted the 100 systems_prompt instantiated to LLMs GPT-4o mini and Llama (200 total responses)
- The LLMs recommended 483 distinct types of foods to the 100 patients,
  - Of these it was possible to validate 178 foods (36.8%) with FoodKG
- Considering the sub-group of 30 diabetic patients:
  - LLMs recommended 260 distinct types of foods
  - It was possible to validate 123 foods (47.3%) with FoodKG.
- **Diabetic Fails:** The LLMs recommended 10 distinct foods with high GI >= 50 to diabetic patients: "brown rice" (GI=50), "crackers"(GI=80), "honey"(GI=60), "kiwi"(GI=50), "mango"(GI=56), "maple syrup"(GI=65), "pineapple"(GI=66), "popcorn"(GI=55), "rice cake"(GI=85), and "watermelon" (GI=75)

Description	Llama3.1-8B	GPT-4o mini	Total	%
All patients (100)				
Distinct food recommendations	315	308	483	
Coverage of FoodKG	135	122	178	36.8%
Diabetic patients (30)				
Distinct food recommendations	172	164	260	
Coverage of FoodKG	97	80	123	47.3%
Foods with GI >=50	6	. 6	10	8.3%

ChatBS-NexGen Nutrition Case Study: Diabetic "fails"

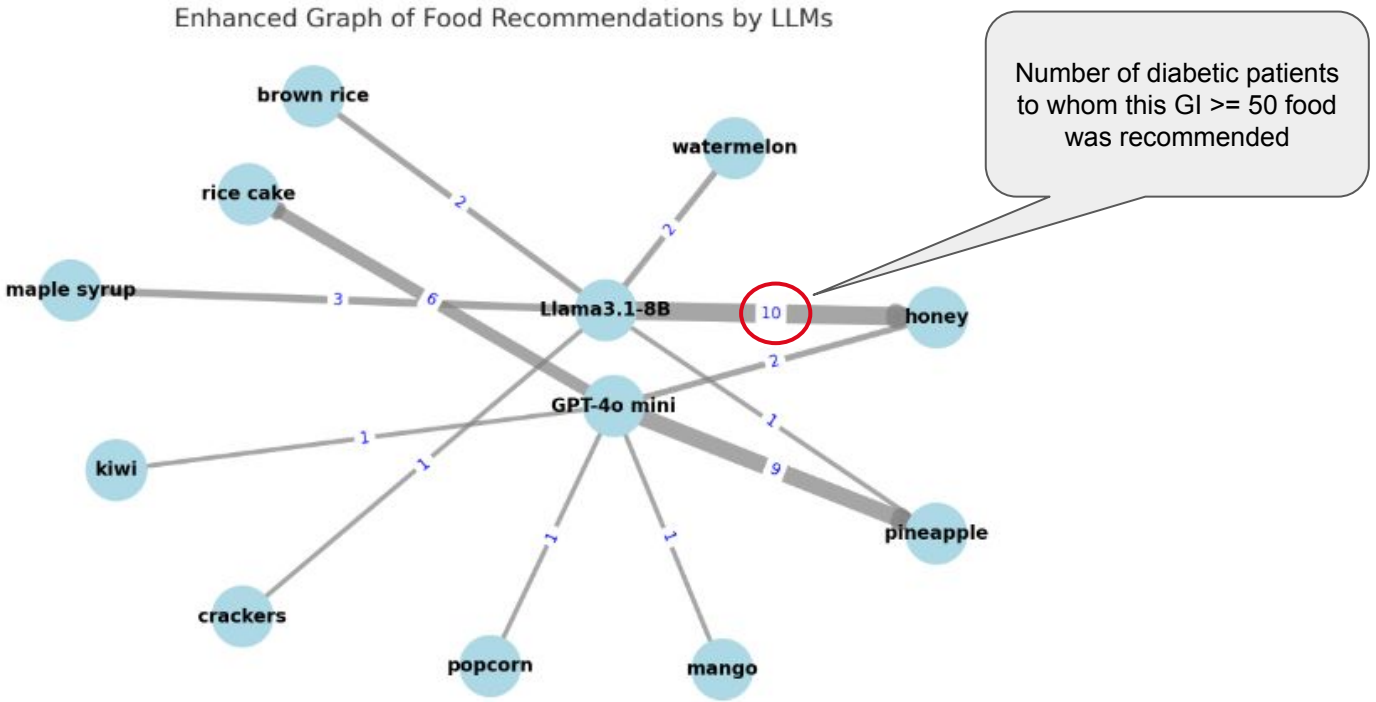


Figure 2: A Graph with High Glycemic Index Foods (GI >= 50) recommended by LLMs: Llama3.1-8B vs. GPT-4o Mini

ChatBS-NexGen KG Experiment: KG Takeaways

- A significant KG issue is the incompleteness of the data.
  - While KGs contained certain food items, many were either missing entirely or lacked critical attributes such as glycemic index (GI) values.
  - This gap in data coverage constrains the applicability of KGs for comprehensive analyses.
- Another limitation is the query formulation for domain-specific applications
  - This did not present a direct challenge in the nutrition use case, but…
  - The process of designing esp. SPARQL queries to extract relevant data from KGs might be a problem in more general and/or less structured scenarios.
- These limitations underline the necessity for advancements in both KG data comprehensiveness and user-friendly, ontology-driven KG (ie SPARQL) query generation mechanisms.

ChatBS-NexGen KG Experiment: Architecture Takeaways

- Our experiment demonstrated the potential benefits of integrating KGs into a structured framework for managing LLM experimentation workflows.
- Through 100 prompts resubmitted 10 times to two LLMs, it was possible to identify inappropriate recommendations and high variability in responses, considering 10 repetitions.
- The proposed architecture aims to transition from the current state of manual experimentation to a more robust testbed environment for LLM experimentation.
- By leveraging KG-based methodologies, this architecture promises enhanced automation and consistency in handling LLM responses.

Let's build it!

Rensselaer

why not change the world?®

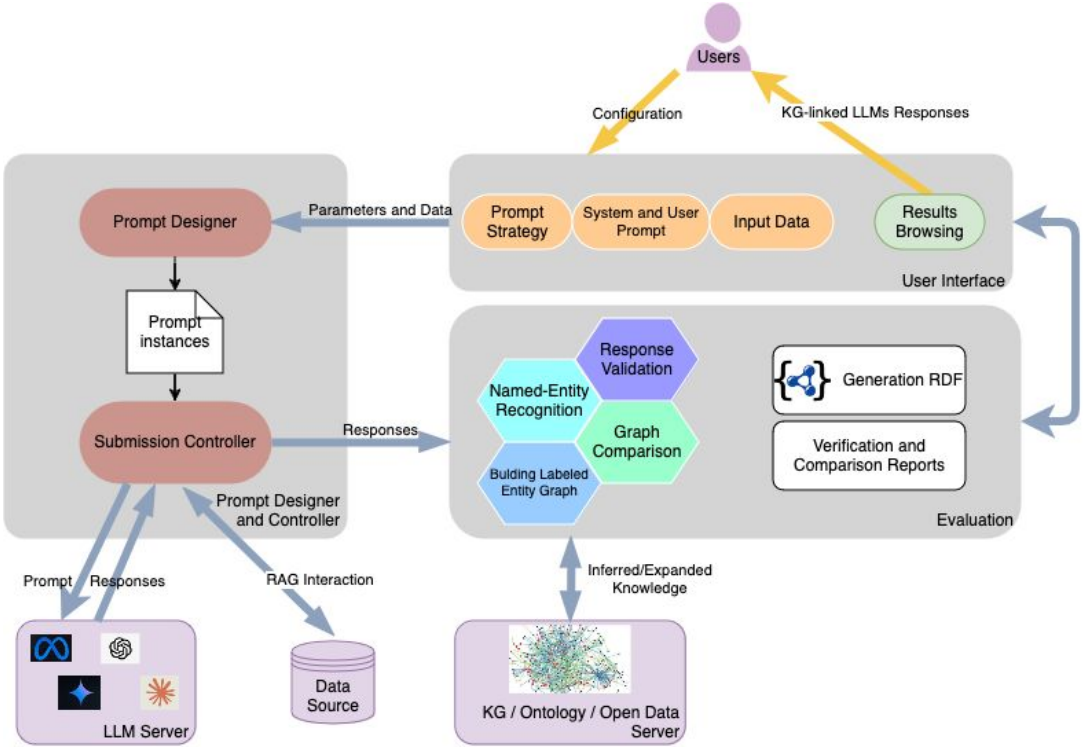
ChatBS-NexGen Architecture

Nipun Pathirage
Devin De Silva
January 2025

The Tetherless World Constellation
Rensselaer Polytechnic Institute



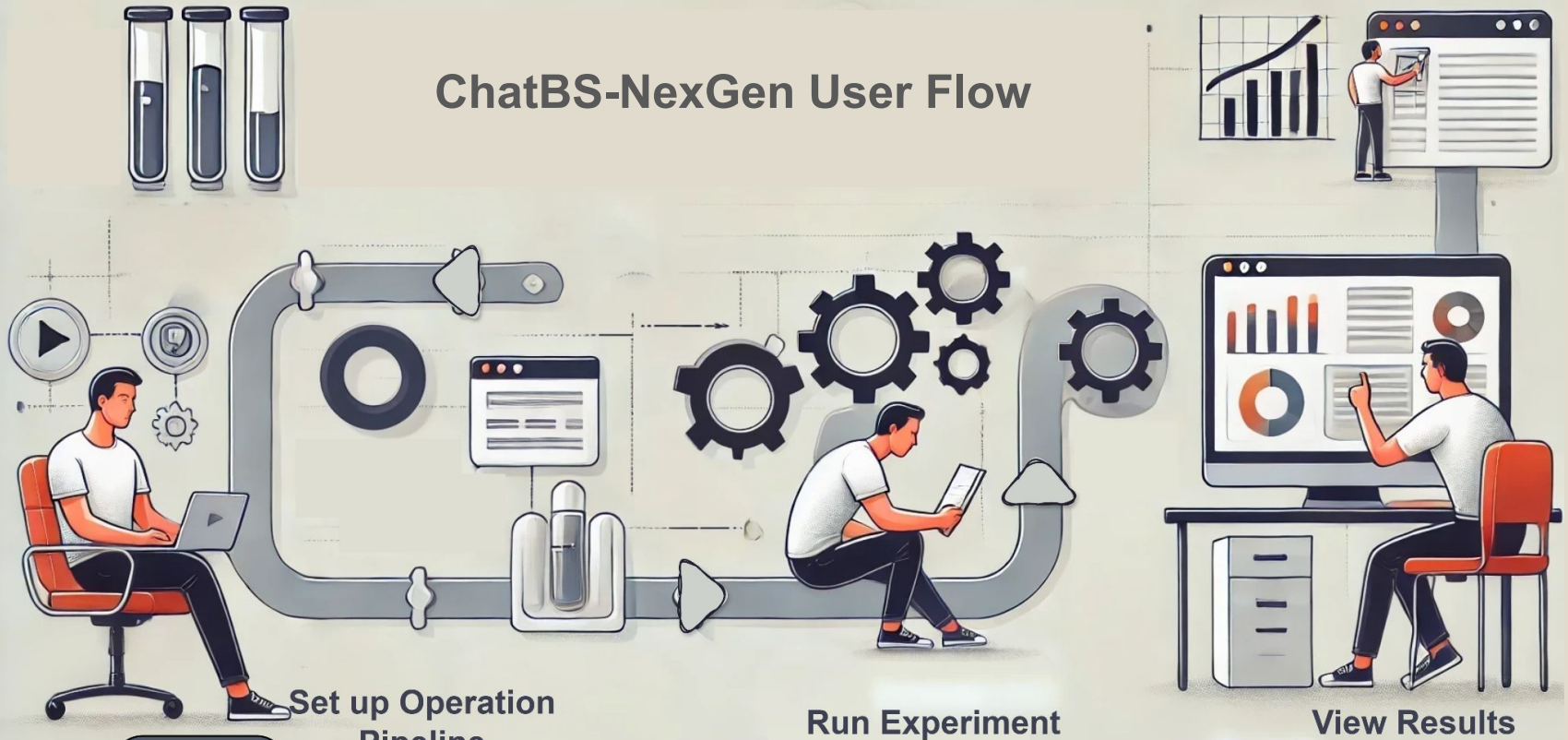
ChatBS-NexGen Architecture (from JoWS paper)



Objectives

- An intuitive platform for validating large language models (LLMs) against authoritative knowledge graphs.
- Effortlessly create text-based experiment pipelines and automate validation processes to ensure factual accuracy and reliability in responses.

ChatBS-NexGen User Flow



Step 1

Set up Operation Pipeline

Step 2

Run Experiment

Step 3

View Results

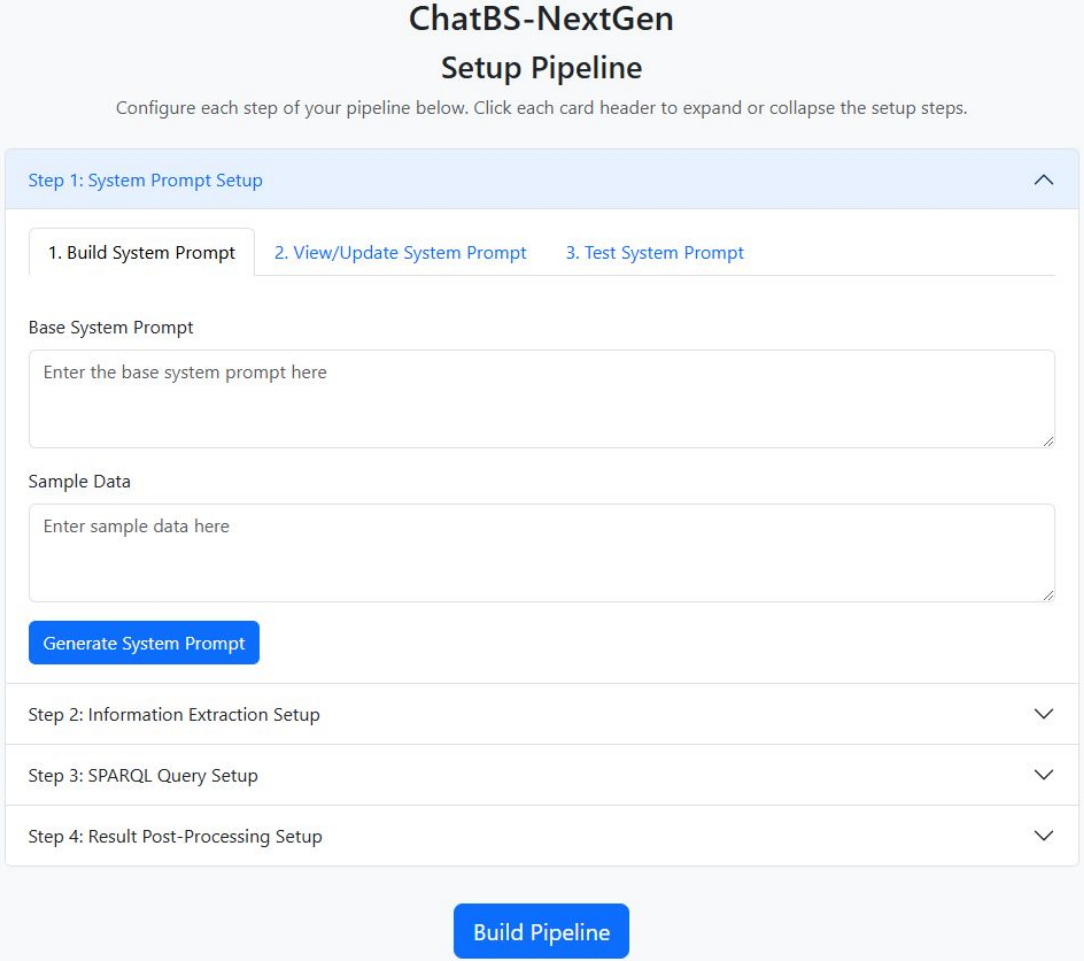
Step 1 : Setup

Process step

a. System Prompt
b. Information Extraction
c. SPARQL Query template
d. Post-Processing

Input : Sample dataset

Output : Generated pipeline



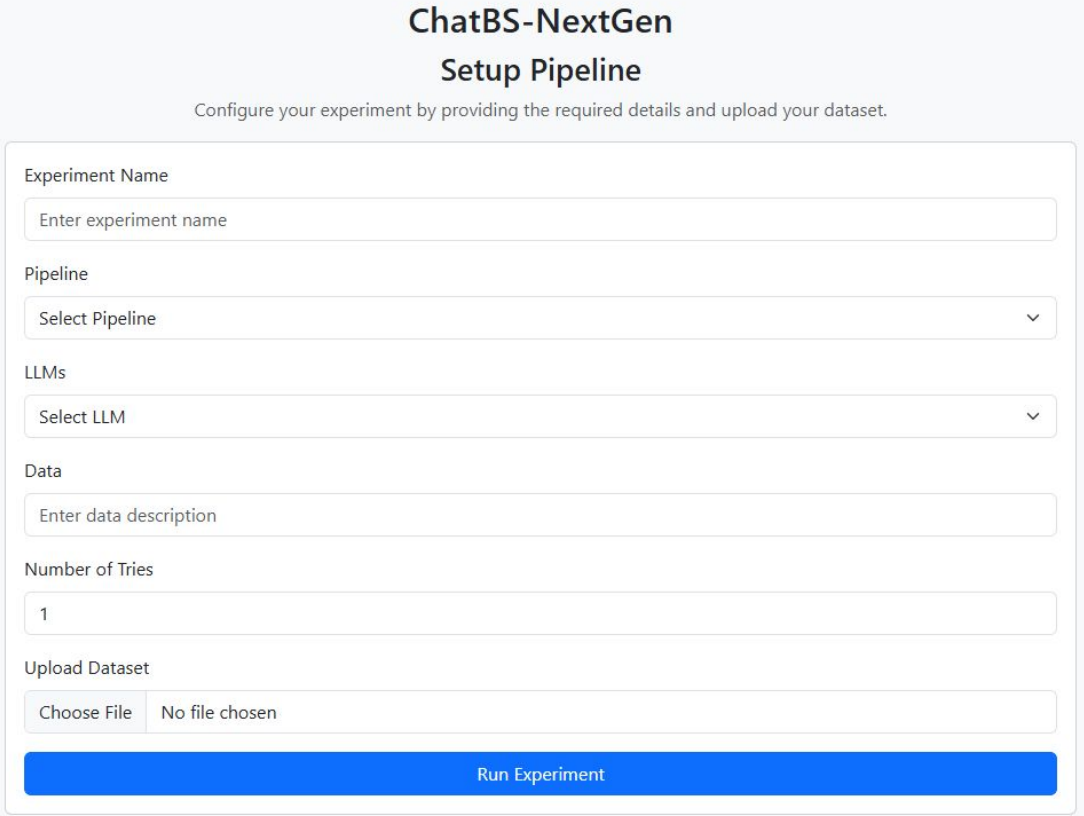
Step 2 : Run

Process step

a. Configure Experience
b. Run & View Experience

Input :
Experiment configs
Experiment dataset

Output :
Experiment run logs

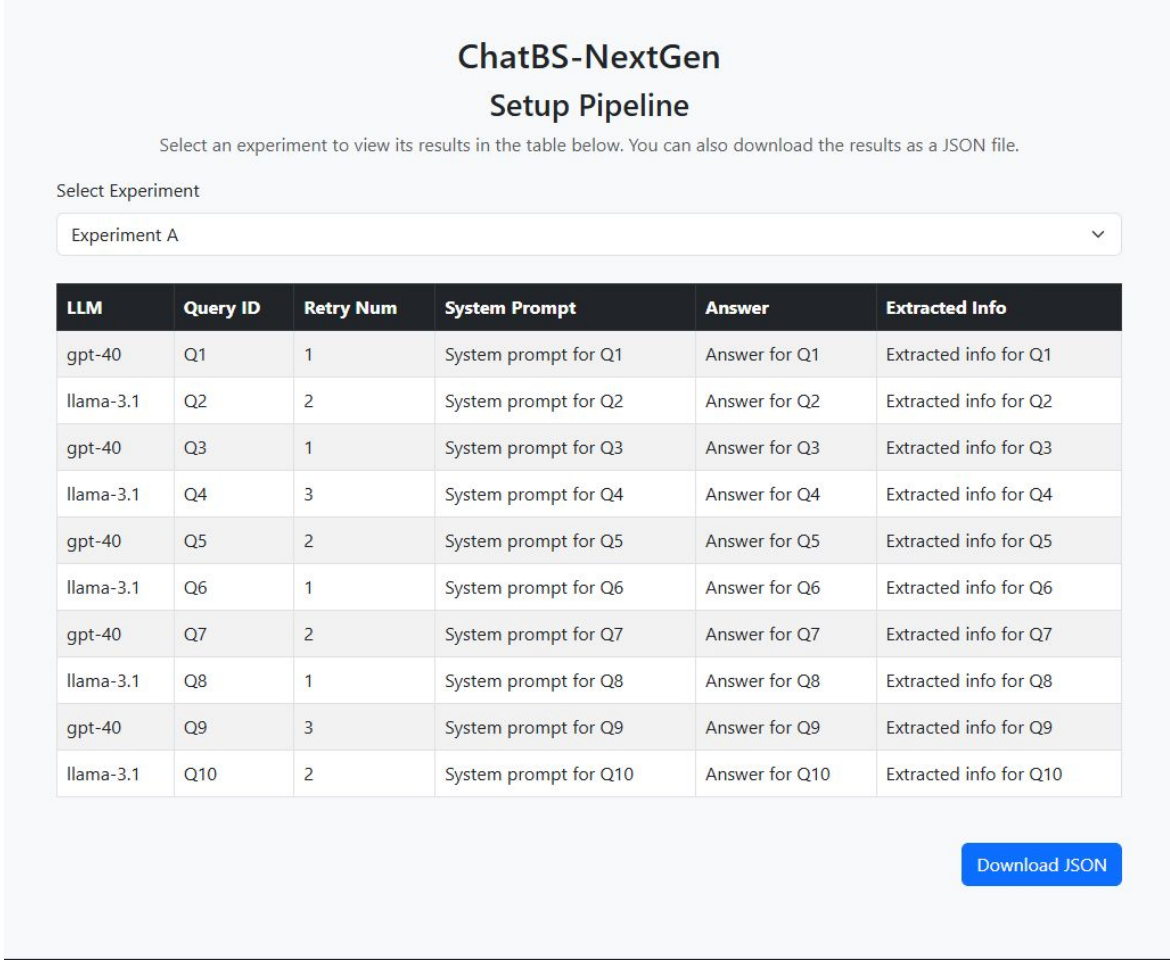


Step 3 : View

Process step

a. View Experiment Results

Output :
JSON experiment results



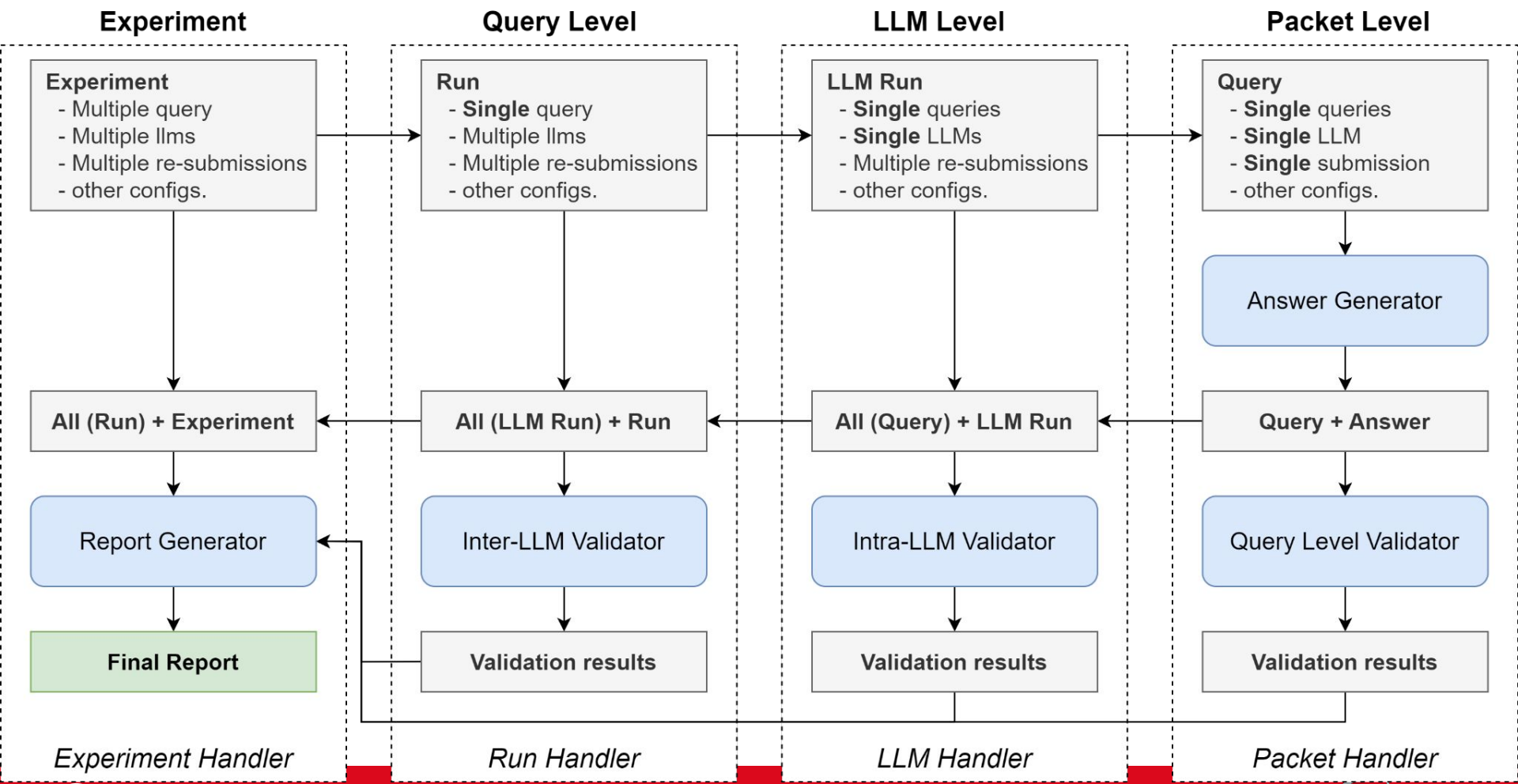
LLM	Query ID	Retry Num	System Prompt	Answer	Extracted Info
gpt-40	Q1	1	System prompt for Q1	Answer for Q1	Extracted info for Q1
llama-3.1	Q2	2	System prompt for Q2	Answer for Q2	Extracted info for Q2
gpt-40	Q3	1	System prompt for Q3	Answer for Q3	Extracted info for Q3
llama-3.1	Q4	3	System prompt for Q4	Answer for Q4	Extracted info for Q4
gpt-40	Q5	2	System prompt for Q5	Answer for Q5	Extracted info for Q5
llama-3.1	Q6	1	System prompt for Q6	Answer for Q6	Extracted info for Q6
gpt-40	Q7	2	System prompt for Q7	Answer for Q7	Extracted info for Q7
llama-3.1	Q8	1	System prompt for Q8	Answer for Q8	Extracted info for Q8
gpt-40	Q9	3	System prompt for Q9	Answer for Q9	Extracted info for Q9
llama-3.1	Q10	2	System prompt for Q10	Answer for Q10	Extracted info for Q10

ChatBS Current Features

- User and system prompt direct input
- LLM selection (multiple LLMs)
- Prompt engineering technique [CoT and reference generation]
- Prompt re-submissions
- View & download the generated answers
- Generate Google search link for all the generated sentences
- RDF relationship generation from the generated answer
- View as table and download in JSON-LD serialized format

ChatBS-NexGen New Features

- Support for multiple query (user & system) submission at a time.
- Automatic answer validation at:
  - QA level
  - LLM level (intra-LLM validation)
  - Experiment level (inter-LLM validation)
- Generate intuitive analysis report


Experiment
Query Level
LLM Level
Packet Level
Experiment
- Multiple query
- Multiple llms
- Multiple re-submissions
- other configs.
Run
- Single query
- Multiple llms
- Multiple re-submissions
- other configs.
LLM Run
- Single queries
- Single LLMs
- Multiple re-submissions
- other configs.
Query
- Single queries
- Single LLM
- Single submission
- other configs.
Answer Generator
All (Run) + Experiment
All (LLM Run) + Run
All (Query) + LLM Run
Query + Answer
Report Generator
Inter-LLM Validator
Intra-LLM Validator
Query Level Validator
Final Report
Validation results
Validation results
Validation results
Experiment Handler
Run Handler
LLM Handler
Packet Handler

Fact Validation: Custom GI Validation

