Research Task 08 – Initial Planning Report

Date: October 15, 2025

Bias Detection in LLM-Generated Data Narratives

1. Objective & Context

Building upon previous tasks that generated statistical and narrative insights for the Syracuse Women's Lacrosse 2024 dataset, this phase aims to design a controlled experiment that tests whether Large Language Models (LLMs) exhibit systematic bias when interpreting identical data under differently framed prompts. The experiment will measure how prompt framing—positive vs negative tone, demographic references, and goal orientation—affects narrative outcomes such as player recommendations and sentiment.

2. Datasets & Sources

The same anonymized sports datasets used in Task 05 will be reused: player_stats.csv, team_stats.csv, period_stats.csv, and game_schedule.csv. All personally identifying names will be replaced with neutral identifiers ('Player A', 'Player B', etc.) following ethical guidelines.

3. Initial Hypotheses

ID	Hypothesis	Bias Type
H1	Narratives describing a player as 'struggling' vs 'developing' will yield different recommendations.	Framing Bias
H2	Prompts with demographic hints (class year or position) will alter 'potential' labeling.	Demographic Bias
H3	Asking 'what went wrong' vs 'what opportunities exist' will shift response sentiment.	Confirmation/Sentiment Bias

4. Planned Experimental Design

Models under test include GPT-4, Claude, and Gemini. Each hypothesis will have two minimally different prompt variants. Each prompt embeds numeric stats to ensure factual grounding. Responses will be collected 3–5 times per prompt per model and logged (prompt, timestamp, model version, response) for later sentiment and keyword analysis.

5. Tools & Setup

Python, Pandas, TextBlob, and VADER will be used for sentiment and keyword analysis. LLM querying will use OpenAI API, Claude.ai, and Gemini. A GitHub repository (Task_08_Bias_Detection) will store scripts.

6. Ethical & Compliance Checklist

- All player names replaced with anonymous identifiers.
- No PII stored or uploaded to GitHub.
- All AI outputs clearly labeled as synthetic.
- Reproducibility ensured by pinning model versions and random seeds.

7. Next Steps

1. Refine prompt templates and store in /prompts folder.

2. Write experiment_design.py for automatic prompt generation.

3. Prepare sentiment and keyword detection functions in Python.