CHUNG-YEH(OLIVER) YANG

(646) 992-7153 | cy2816@columbia.edu | linkedin.com/in/olivery0307 | https://github.com/Olivery0307

EDUCATION

Columbia University New York, NY
M.S. in Data Science Dec 2026(Expected)
Relevant Coursework: Exploratory data analysis, Computer Vision, Operational Logistics, Machine Learning, Poisson Process

Boston University Boston, MA
B.S. in Data Science, GPA: 3.8/4.0 Sep 2020 - Dec 2024
Major in Data Science, Minor in Business Administration
Coursework: Bayesian Statistics, Computer Systems, Database Design, Data Structure Algorithms, Deep Learning, Hypothesis Testing, Linear Algebra, Machine Learning, Multivariable Calculus

WORK EXPERIENCE

PalAI New York, NY
AI/ML Intern Sep 2025 – Present
- Engineered and prototyped a core ML and AI-driven college application and essay feedback platform
- Designed a data pipeline to standardize and augment 100+ essay/feedback pairs and fine-tuning the model with the Gemini API
- Collaborated with cross-functional backend and frontend engineers to deploy ML models into a production-level user experience

Boston University Questrom School of Business Boston, MA
Research Assistant in Applied AI in Business Aug 2023 - Jan 2025
- Collaborated with 5+ academic researchers to design and implement four AI solutions tailored to specific business challenges
- Analyzed over 50,000 user queries to measure the impact of query characteristics on RAG generation and user behaviors
- Engineered a pipeline that uses an LLM to label artwork genres from unsupervised clusters, achieving zero-shot topic modeling

BU Spark! Boston, MA
Data Science Technical Project Manager Sep 2024 - Dec 2024
- Managed six data analysis and visualization projects, coordinating with clients and facilitating student communication
- Provided technical assistance and leadership on data visualization projects leveraging Python, Tableau, and Google BigQuery
- Configured dataset and software setup for Notion and GitHub, ensuring teams had resources for project milestones

GIGA RESET Shanghai, CN
Data Science Summer Intern Jun 2024 - Aug 2024
- Designed a data processing pipeline for the RESET Air project with Python, reducing manual workload by 80%
- Automated creation of scorecards by importing Python libraries, optimizing slide generation and data visualization

PROJECTS

YouBike Prediction Service Jun 2025 - Present
- Developed a serverless data pipeline on AWS (Lambda, S3) to collect and process real-time YouBike API data
- Engineered time-series features to train a LightGBM model, achieving over 90% recall in predicting critical station states
- Deployed and dockerized and prediction pipeline using AWS Lambda and ECR for automated, live inference

Deep Neural Networks Fusion Approach for VQA Verification Oct 2024 - Dec 2024
- Built a custom preprocessing pipeline with Pytorch to balance and label correct and incorrect answers for effective training
- Constructed a fusion-based model combining image and text features to classify answers in Visual Question Answering
- Achieved 87.07% validation accuracy using attention mechanisms, showcasing strong generalization of model

Multi-Modal Topic Modeling for Artworks Jan 2024 - Jul 2024
- Established a zero-shot topic modeling pipeline to identify and label thematic genres in a dataset of unlabeled artworks
- Implemented clustering utilizing CLIP embeddings and applied HDBSCAN to identify 15 distinct thematic clusters
- Pioneered a novel labeling technique by prompting a LLM with images to generate accurate, human-readable genre descriptions

LEADERSHIP EXPERIENCE

Taiwanese Overseas Students Association at Boston University Boston, MA
President Apr 2023 - May 2024
- Organized events and promote Taiwanese culture for over 300 Taiwanese students and general members in Boston University
- Coordinated and distributed work to 20 executive board members, ensuring efficient and event planning and execution

SKILLS

Programming Languages: Python, SQL, R, Rust
ML Libraries & Frameworks: PyTorch, LLM, OpenAI API, LangChain, Transformers, Scikit-Learn
Platforms & Developer Tools: AWS, Docker, Git, Google BigQuery, Tableau, Redis

Quantifying Query Specificity in RAG-based Search Engine

This research is part of an ongoing project I contributed to at Boston University's BITLAB, led by Professor Dokyun Lee. We collaborated with a media company that is testing a new Retrieval-Augmented Generation (RAG) system. For their experiment, users were split into "control" and "treatment" groups, where the treatment group's queries were reformulated by GenAI.

1. The Research Question

My research sought to answer two primary questions:

1. Is there a statistically significant difference in specificity between user-submitted queries (control) and GenAI-reformulated queries (treatment)?
2. Can we design a reliable, corpus-independent metric to quantify a query's "specificity" before a search is executed to predict its ambiguity and potential performance?

This project tested the hypothesis that a GenAI-based search reformulator (the "treatment") would produce queries with demonstrably higher semantic specificity than the original user-submitted queries (the "control").

2. Data

The dataset provided was a large-scale, anonymized search log from the media company's RAG system experiment, delivered as a single parquet file containing over 8.9 million total query events and 34,209 unique query strings. The file included three features: `normalized_query`, `group`, and `time`. All query strings were pre-normalized (e.g., all lowercase, punctuation removed, and words underscore-separated). The data was pre-segmented into the two distinct groups from the A/B test: the Control Group (user-submitted) contained 7,848,077 events, and the Treatment Group (GenAI-reformulated) contained 1,064,032 events.

3. Approach and Methodologies

<u>Part A: Developing the Specificity Construct</u>

Developing this construct was challenging for two reasons: first, traditional metrics like query length are unreliable proxies for specificity; second, published literature often lacks direct, pre-retrieval metrics for quantifying specificity. Therefore, I designed a new, two-part composite score system based on an evidence-gathering process from highly-cited academic papers in information systems and semantic analysis. This score contains two measurements highly correlated with specificity: Named Entity Count and Entity Granularity.

The rationale for Named Entity Count is that entities act as semantic constraints. A query with more entities (e.g., "Apple Park cupertino") is inherently more specific than one with fewer (e.g., "tech companies"). To measure this, I used an LLM to perform Named Entity Recognition (NER)

on all 34,209 unique queries, extracting and classifying entities (e.g., PERSON, LOCATION, ORGANIZATION).

Entity Granularity stands for the intrinsic specificity of an individual entity. The reason this is needed as a supplement to entity count is that not all entities are equal; a query with two low-granularity entities (e.g., "tech companies california") is far less specific than a query with two high-granularity entities (e.g., "Apple Park cupertino"). To measure this, I queried large-scale Knowledge Graphs (KGs) using a multi-KG approach to select the best source for each entity type. For example, for LOCATIONs, I used the GeoNames API, scoring specificity based on the entity's `featureCode` (e.g., a city is more granular than a country).For MISC (abstract concepts), I used WordNet (via NLTK) to measure conceptual depth. Finally, a query's granularity score was aggregated by computing the mean granularity of all its entities.

Part B: Statistical Analysis (Hypothesis Testing)

My hypothesis was that the Treatment group's queries would have significantly higher scores for both entity count and granularity score.

- **Assumption Checking:** The data was count-based, non-normally distributed, and had unequal variances. This ruled out a parametric t-test.
- **Primary Statistical Test:** The non-parametric Mann-Whitney U test was selected as the appropriate method to compare the distributions of the two independent (Control vs. Treatment) groups.
- **Analysis:** I ran two separate Mann-Whitney U tests: one comparing the entity count distributions and one comparing the granularity score distributions.

4. My Findings

The analysis confirmed the hypothesis with high statistical significance. For one, the GenAI reformulation significantly increased the number of entities. The Treatment group had a mean of 0.93, compared to the Control group's 0.79. This difference was highly statistically significant. For another, the tool significantly increased the semantic granularity. The Treatment group had a higher mean granularity score (0.04) than the Control group (0.03). This was also highly statistically significant. In conclusion, the GenAI-reformulated queries were demonstrably more specific, and more semantically constrained than the original user queries. The tool successfully transformed vague queries into more actionable, entity-grounded inputs.

5. Computational Environment

- Programming Language: Python; Report: PowerPoint, Latex(Overleaf)
- Main Libraries:
  - Data Analysis & Manipulation: `pandas`, `numpy`
  - Statistical Analysis: `scipy.stats` for `mannwhitneyu`, `levene`) `statsmodels`
  - Data Visualization: `matplotlib`, `seaborn`
  - API/KG Access: `requests` (for GeoNames/Wikidata APIs), `google-generative-ai` (for NER), `nltk` (for WordNet)