**How I Automated Data Annotation with LLMs (and Why It’s Awesome)**

In the world of AI and data science, automating tasks is a game-changer, especially when it comes to data annotation. For this project, I set out to automate the categorization of research papers scraped from the NeurIPS conference website using a powerful tool—Google Gemini, a Large Language Model (LLM).

**The Process: From Scraping to Classification**

I began by scraping metadata (titles, authors, abstracts) from NeurIPS papers using Python and tools like aiohttp and BeautifulSoup. Once I had the data, it was time to classify each paper into one of these categories: **Deep Learning, Computer Vision, Reinforcement Learning, NLP, and Optimization**.

Here’s where Google Gemini came in. I sent each paper’s title and abstract to the Gemini API and asked it to classify the paper. It then returned a category, and I stored that info in a CSV file.

**Challenges and How I Tackled Them**

1. **Scraping Issues**: Handling missing data and managing timeouts for large amounts of information was tricky, but I worked around it by implementing robust error handling.
2. **API Classifications**: I had to make sure the API responses were accurate and handled any potential errors, like rate limits or timeouts. A bit of tweaking with prompts made all the difference.
3. **Storing Results**: Updating the CSV file without any hiccups required attention to detail, especially with handling large-scale data.

**Results: What I Learned**

* **Time-Saving**: Automating the annotation saved tons of time that would otherwise be spent manually classifying papers.
* **Scalability**: I now have a system that can handle tons of new papers without breaking a sweat.
* **A Structured Dataset**: With papers categorized, I can now easily analyze research trends and categorize papers by topic.

**Key Takeaways**

1. **Automating Annotation**: Using LLMs like Google Gemini for text classification is super powerful and efficient.
2. **Challenges**: From scraping hiccups to fine-tuning the API, there were definitely some challenges, but it was worth it.
3. **Future Improvements**: I plan to add more categories and refine the prompts to get even better results.

In the end, I was able to automate the annotation process, saving time and organizing the research papers in a way that’s useful for deeper analysis. If you’re working on a similar project, using LLMs for classification can make your life a whole lot easier!