JOHNS HOPKINS
CAREY BUSINESS SCHOOL

Lecture 5 Part I

BU.330.760 Generative AI for Business

Minghong Xu, PhD.
Associate Professor

Reflections

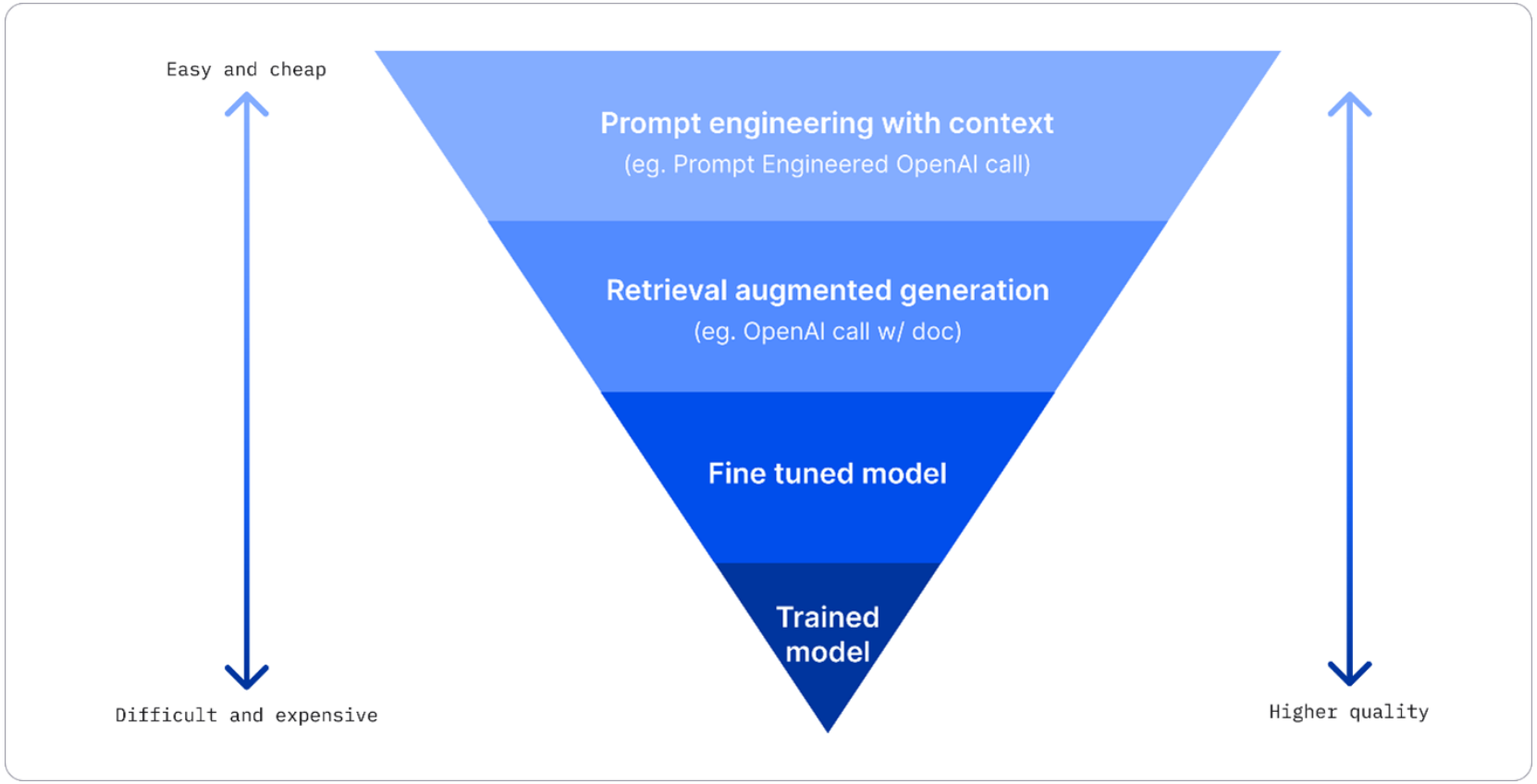
- Agentic AI: “automation”
  - A few “chatGPT”s talking to each other
  - Each handle a specific task
  - Prompt engineering is the core, like HR training the new employee
- Business case
  - Focus: understand a process and automate it

Today's Agenda

- BloombergGPT vs RAG
  - Last part of "Capability"
- LLM risks, governance, adversarial attacks
  - Vulnerability

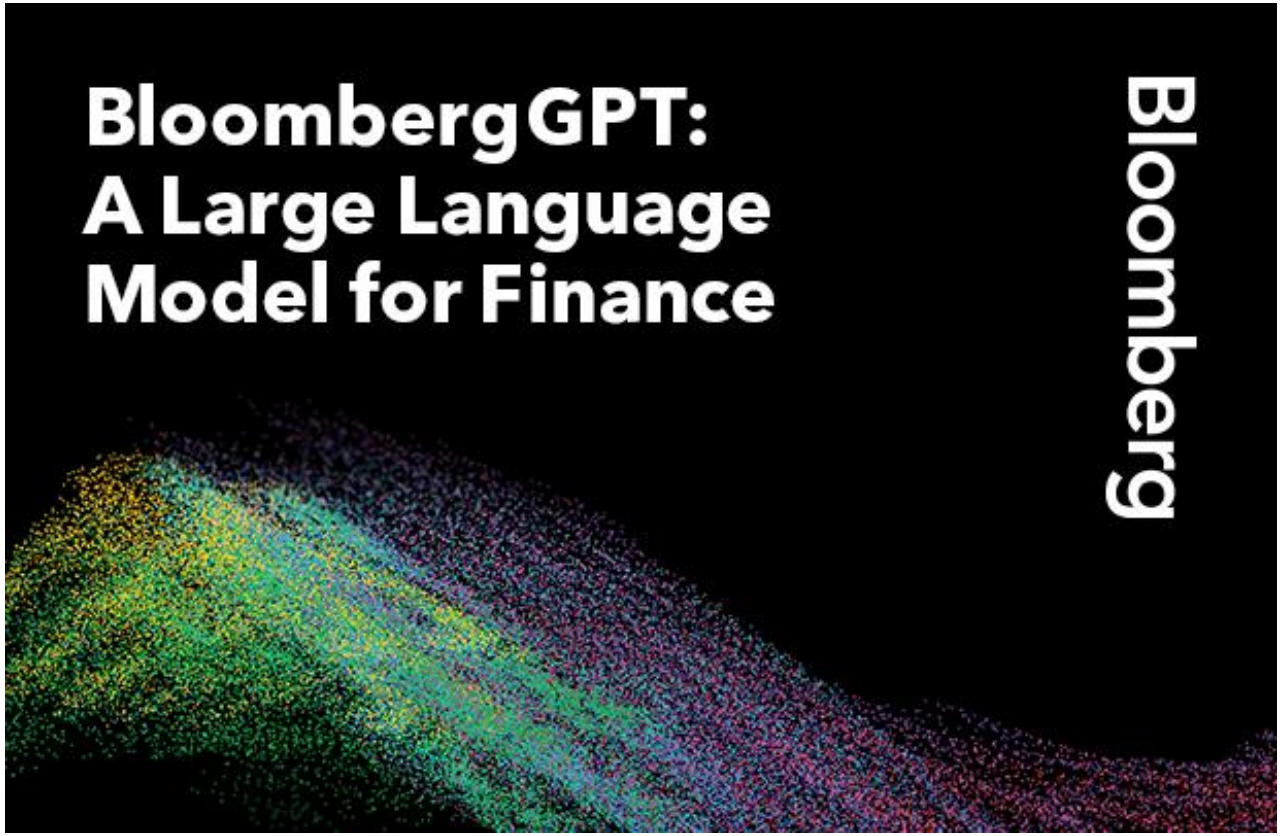
LLM Challenges (Recap)

- Reasoning
  - LLMs are good at pattern recognition and statistical analysis, mimic patterns from training data
  - Lack true understanding, lack arithmetic, commonsense, and symbolic reasoning
- Domain adaption
  - Incorporate domain knowledge and local expertise



[Four Ways that Enterprises Deploy LLMs | Fiddler AI Blog]

BloombergGPT



» https://arxiv.org/abs/2303.17564

Model Overview

- 50 billion parameters
- Combining general-purpose LLM and domain-specific LLM
- Financial language documents over 40 years
- 700 billion tokens

Dataset	Docs 1e4	C/D	Chars 1e8	C/T	Toks 1e8	T%
FINPILE	*175,886*	*1,017*	*17,883*	*4.92*	*3,635*	*51.27%*
Web	158,250	933	14,768	4.96	2,978	42.01%
News	10,040	1,665	1,672	4.44	376	5.31%
Filings	3,335	2,340	780	5.39	145	2.04%
Press	1,265	3,443	435	5.06	86	1.21%
Bloomberg	2,996	758	227	4.60	49	0.70%
PUBLIC	*50,744*	*3,314*	*16,818*	*4.87*	*3,454*	*48.73%*
C4	34,832	2,206	7,683	5.56	1,381	19.48%
Pile-CC	5,255	4,401	2,312	5.42	427	6.02%
GitHub	1,428	5,364	766	3.38	227	3.20%
Books3	19	552,398	1,064	4.97	214	3.02%
PubMed Central	294	32,181	947	4.51	210	2.96%
ArXiv	124	47,819	591	3.56	166	2.35%
OpenWebText2	1,684	3,850	648	5.07	128	1.80%
FreeLaw	349	15,381	537	4.99	108	1.52%
StackExchange	1,538	2,201	339	4.17	81	1.15%
DM Mathematics	100	8,193	82	1.92	43	0.60%
Wikipedia (en)	590	2,988	176	4.65	38	0.53%
USPTO Backgrounds	517	4,339	224	6.18	36	0.51%
PubMed Abstracts	1,527	1,333	204	5.77	35	0.50%
OpenSubtitles	38	31,055	119	4.90	24	0.34%
Gutenberg (PG-19)	3	399,351	112	4.89	23	0.32%
Ubuntu IRC	1	539,222	56	3.16	18	0.25%
EuroParl	7	65,053	45	2.93	15	0.21%
YouTubeSubtitles	17	19,831	33	2.54	13	0.19%
BookCorpus2	2	370,384	65	5.36	12	0.17%
HackerNews	82	5,009	41	4.87	8	0.12%
PhilPapers	3	74,827	23	4.21	6	0.08%
NIH ExPorter	92	2,165	20	6.65	3	0.04%
Enron Emails	24	1,882	5	3.90	1	0.02%
Wikipedia (7/1/22)	2,218	3,271	726	3.06	237	3.35%
TOTAL	*226,631*	*1,531*	*34,701*	*4.89*	*7,089*	*100.00%*

Table 1: Breakdown of the full training set used to train BLOOMBERGGPT. The statistics provided are the average number of characters per document ("C/D"), the average number of characters per token ("C/T"), and the percentage of the overall tokens ("T%"). Units for each column are denoted in the header.

Training Data

- Financial Datasets
  - 363B tokens, 51.27% of training
  - Created internally or acquired from external
  - Time-stamped from 2007-3-1 to 2022-7-31
  - Cleaning: remove markup, special formatting and templates
  - Types
    - Web: crawl from quality websites containing financially relevant information
    - News: all new articles, including transcripts of Bloomberg TV news
    - Filings: financial statements, 10-K and 10-Q from EDGAR
    - Press: press releases issued by companies, public communications similar to news stories
    - Bloomberg: Bloomberg authored news, opinions and analyses

Training Data (Cont.)

- Public Datasets
  - 345B tokens, 48.73% of training
  - The Pile: dataset used in GPT-Neo (Black et al., 2021), GPTJ (Wang and Komatsuzaki, 2021), and GPT-NeoX (20B) (Black et al., 2022)
  - Colossal Clean Crawled Corpus (C4): dataset used in training T5 (Raffel et al., 2020)
  - Wikipedia: dump of English Wikipedia from July 1, 2022

Model Specifications

- Decoder-only
- 70 layers of transformer decoder blocks
- Budget: 1.3M GPU hours on 40GB A100
- Use PyTorch, and Amazon SageMaker service
  - 64 p4d.24xlarge instances

Shape	
Number of Layers	70
Number of Heads	40
Vocabulary Size	131,072
Hidden Dimension	7,680
Total Parameters	50.6B
Hyperparameters	
Max Learning Rate	6e-5
Final Learning Rate	6e-6
Learning Rate schedule	cosine decay
Gradient Clipping	0.3
Training	
Tokens	569B
Hardware	64×8 A100 40GB
Throughput	32.5 sec/step
avg. TFLOPs	102
total FLOPS	2.36e23

Table 4: A summary of the hyper-parameters and their values for BloombergGPT.

Evaluation

Suite	Tasks	What does it measure?
Public Financial Tasks	5	Public datasets in the financial domain
Bloomberg Financial Tasks	12	NER and sentiment analysis tasks
Big-bench Hard (Suzgun et al., 2022)	23	Reasoning and general NLP tasks
Knowledge Assessments	5	Testing closed-book information recall
Reading Comprehension	5	Testing open-book tasks
Linguistic Tasks	9	Not directly user-facing NLP tasks

Table 5: Evaluation Benchmarks. We evaluate BLOOMBERGGPT on a high-coverage set of standard benchmarks that assess downstream performance, taken from HELM, SuperGLUE, MMLU, and the GPT-3 suite. Since these have significant overlap and/or include each other, we restructure them into the categories presented here. We only evaluate on one setup per dataset. We further assess BLOOMBERGGPT on a suite of internal and public financial tasks.

Financial Tasks

- Some NLP tasks can be different on financial language

"COMPANY to cut 10,000 jobs"

- *Is it positive or negative in sentiment?*

Task	Template/Example
Discriminative	
Sentiment Analysis	{sentence} Question: what is the sentiment? Answer: {negative/neutral/positive}
Aspect Sentiment Analysis	{sentence} Question: what is the sentiment on {target}? Answer: {negative/neutral/positive}
Binary Classification	{sentence} Question: {question}? Answer: {Yes/No}
Generative	
NER	Steve Jobs is the CEO of Apple Extract named entity: Steve Jobs (person), Apple (organization)
NER+NED	AAPL stopped using Intel Chips Extract ticker: AAPL, INTC
QA	{context} Question: {question}? Answer: {answer}

Table 7: Template for the different tasks we evaluate in the financial domain.

General NLP Tasks

- Knowledge assessment
  - Ability to recall information seen in training, e.g. science and commonsense
- Reading comprehension
  - Ability to generate correct responses given input text
- Basic understanding of language itself
  - Evaluate disambiguation
  - Grammar check
  - Recognize entailment
  - Words in context

Benefits of Domain Specialization

- Generation of Bloomberg query language (BQL)
  - Translate natural language queries into valid BQL

Input: Get me the last price and market cap for Apple
Output: `get(px_last,cur_mkt_cap) for(['AAPL US Equity'])`

Input: Tesla price
Output: `get(px_last) for(['TSLA US Equity'])`

Input: Get the yield and spread for EC527035 Corp and AL580550 Corp
Output: `get(yield,spread) for(['EC527035 Corp','AL580550 Corp'])`

Input: apple and ibm market cap and eps
Output: `get(cur_mkt_cap,is_eps) for(['AAPL US Equity','IBM US Equity'])`

Input: industry subgroup of ibm apple microsoft google
Output: `get(industry_subgroup()) for(['AAPL US Equity','IBM US Equity', 'MSFT US Equity','GOOGL US Equity'])`

Figure 4: Using BloombergGPT to generate valid Bloomberg Query Language. Using only a few examples in a few-shot setting, the model can utilize its knowledge about stock tickers and financial terms to compose valid queries to retrieve the data, given a request in natural language. In each case, the model is given 3 examples (not shown) followed by the 'Input" and a prompt of "Output:".

Benefits (Cont.)

- Suggestion of news headlines
  - Assist journalists in writing short headlines

Input: The US housing market shrank in value by $2.3 trillion, or 4.9%, in the second half of 2022, according to Redfin. That's the largest drop in percentage terms since the 2008 housing crisis, when values slumped 5.8% during the same period.
Output: Home Prices See Biggest Drop in 15 Years

Input: The global economy is in a better place today than many predicted months ago, Janet Yellen said at the G20. At home, she pointed to a resilient US economy, where headline inflation has moderated and the labor market is strong. She also called for the IMF to move swiftly toward a fully-financed program for Ukraine.
Output: Yellen Sees Global Economy More Resilient Than Expected

Input: Google was sued by the US and eight states seeking the breakup of its ad-tech business for allegedly monopolizing the digital advertising market. The lawsuit is the Biden administration's first big challenge to a tech titan and one of the rare times since 1982 that the DOJ has sought to cleave up a major company.
Output: Google Sued for Monopoly in Online Ad Market

Figure 5: Using BloombergGPT to generate short headline suggestions in a three-shot setting. Bloomberg News sends many newsletters a day that requires these headlines. BloombergGPT could help with the editing process by suggesting initial headlines from the text.

Benefits (Cont.)

- Financial question answering
  - Query financial knowledge, such as CEO of a company

Input: CEO of Assicurazioni Generali?
BloombergGPT: Philippe Donnet
GPT-NeoX: Antonio De Lorenzo, Simone Gambarini, Enrico Zanetti
FLAN-T5-XXL: John M Forsyth, Christopher K Peters, {empty string}

Input: CEO of Silicon Valley Bank?
BloombergGPT: Greg Becker
GPT-NeoX: John Hodor, Mike Fridgen, Peter R Briger
FLAN-T5-XXL: Christian Klein, John M Forsyth, John M Forsyth

Input: CEO of IHS Markit Ltd?
BloombergGPT: Lance Uggla
GPT-NeoX: John F Lechleiter, David Steward, Niall O Broin
FLAN-T5-XXL: William M Hawkins, John M Forsyth, John M Forsyth

Input: CEO of Walgreens Boots Alliance?
BloombergGPT: Rosalind Brewer
GPT-NeoX: Mark V Dei, Stefano Pessina, Stefano Pessina
FLAN-T5-XXL: Christian Klein, John M Forsyth, John M Forsyth

Input: CEO of Citigroup Inc?
BloombergGPT: Jane Fraser
GPT-NeoX: Michael L Corbat, Michael L Corbat, Michael L Corbat*
FLAN-T5-XXL: Christian Sewing, John M Forsyth, John M Forsyth

Figure 6: Testing the ability of BloombergGPT, GPT-NeoX, and FLAN-T5-XXL to recall the names of CEOs of companies. Each model is run in a 10-shot setting. We sample up to three answers and present all of them if they are incorrect. *Michael Corbat was CEO of Citigroup until 2021, highlighting the importance of an up-to-date model.

Ethics and Limitations

- Ensure accurate and factual information is crucial
- Comply with regulations
- Toxicity and bias, harmful language

- Limitations
  - You need to have enough data to train your domain-specific LLM
  - You need sufficient computing resources

JOHNS HOPKINS
CAREY BUSINESS SCHOOL

Retrieval-Augmented Generation

Problems to address

- Pre-trained language models store factual knowledge in parameters
  - Parameterized implicit knowledge base
- *Issues?*
- Memory is “rigid”, difficult to update as parameters are difficult to update
- Hallucinations

A Hybrid Model

- Combine parametric memory and non-parametric memory

Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, Douwe Kiela

- Published in 2021 by researchers from Meta AI
- https://arxiv.org/abs/2005.11401
- Enhance LLMs with your own data, internal knowledge base

Mathematical Model (Optional)

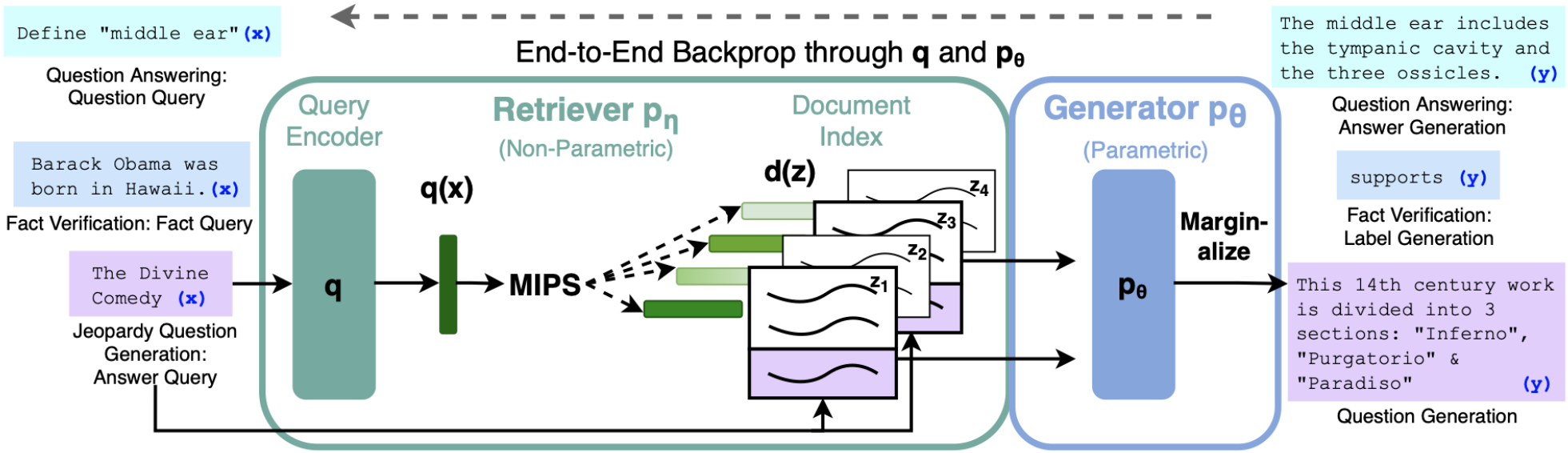


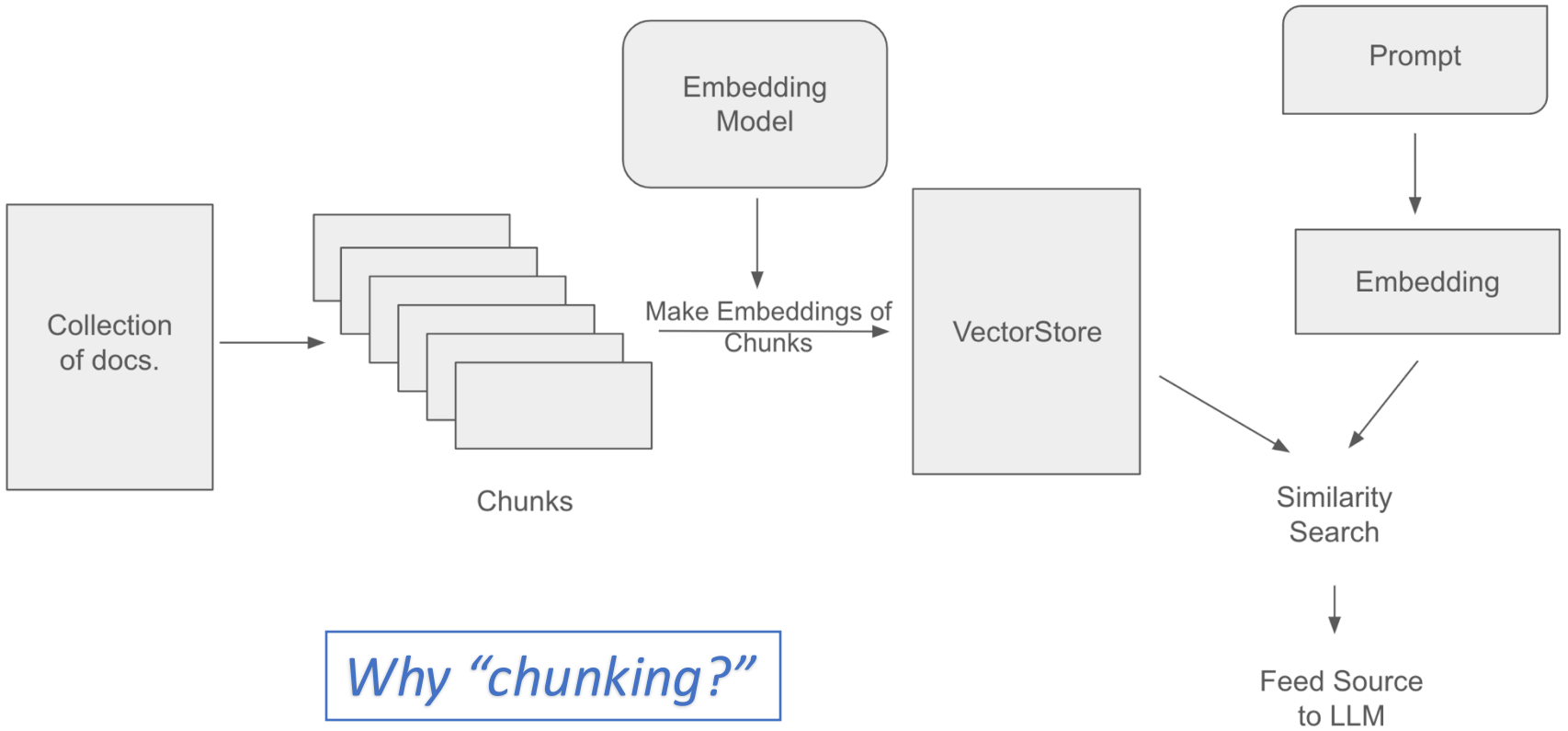
Figure 1: Overview of our approach. We combine a pre-trained retriever (*Query Encoder* + *Document Index*) with a pre-trained seq2seq model (*Generator*) and fine-tune end-to-end. For query x, we use Maximum Inner Product Search (MIPS) to find the top-K documents z_i. For final prediction y, we treat z as a latent variable and marginalize over seq2seq predictions given different documents.

Two Components

- **Retrieval Component**: search a large database or corpus of documents to find relevant information
  - Similar to search engine, find pages related to a search query
  - Vector database to vectorize the “local” knowledge

- **Generation Component**: take the retrieved documents as additional context and generate a response
  - leverage the strength of generative models to produce coherent and contextually appropriate text
  - “Augmented generation”

Architecture



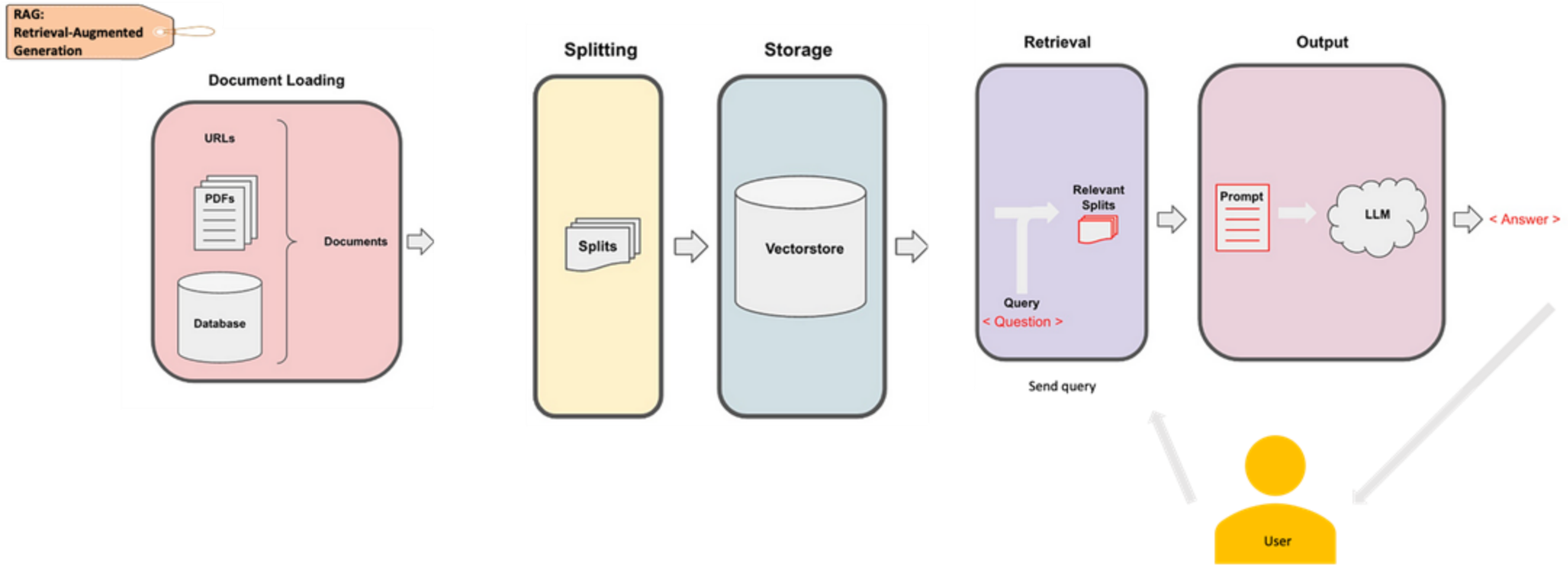
Why "chunking?"

Vector Database

- *What is vectorize (Recap)?*
- Numerical representation of text, embedding

- LLMs have a limit of the number of tokens they accept as input
- Break down into small chunks
- Embeddings most like the question will be returned
- Reduces costs on the number of tokens passed through an LLM

Another View



Experiments

- Non-parametric knowledge source: December 2018 Wikipedia dump
- Split into disjoint 100-word chunks, total 21M documents
- Top 5 or top 10 documents for each query
- Task types
  - Open-domain QA
  - Abstractive QA, e.g. “what is the weather in Baltimore, MD”
  - Jeopardy question generation, trying to guess an entity form a fact, e.g. “In 1986 Mexico scored as the first country to host this international sports competition twice”
  - Fact verification, retrieving evidence and reasoning over the evidence to classify true or false or unverifiable

What Problems does RAG Solve

- Cost-effective implementation
  - Avoids expensive model retraining, makes AI more accessible
  - Cost-effective approach to introducing new data to the LLM
- Current Information
  - Allows developers to provide the latest research, statistics, or news to the generative models by connecting to information sources
- Enhanced user trust
  - Provides source attribution, enables verification
- More developer control
  - Developers can test and improve their chat applications more efficiently
  - Better security

https://aws.amazon.com/what-is/retrieval-augmented-generation/

BloombergGPT vs RAG (Discussion)

Aspects	**BloombergGPT (Domain-specific)**	**RAG**
Training		
Knowledge Base		
Cost		
Flexibility		
Real-time updates		
Deep domain knowledge		

Challenges from Practice

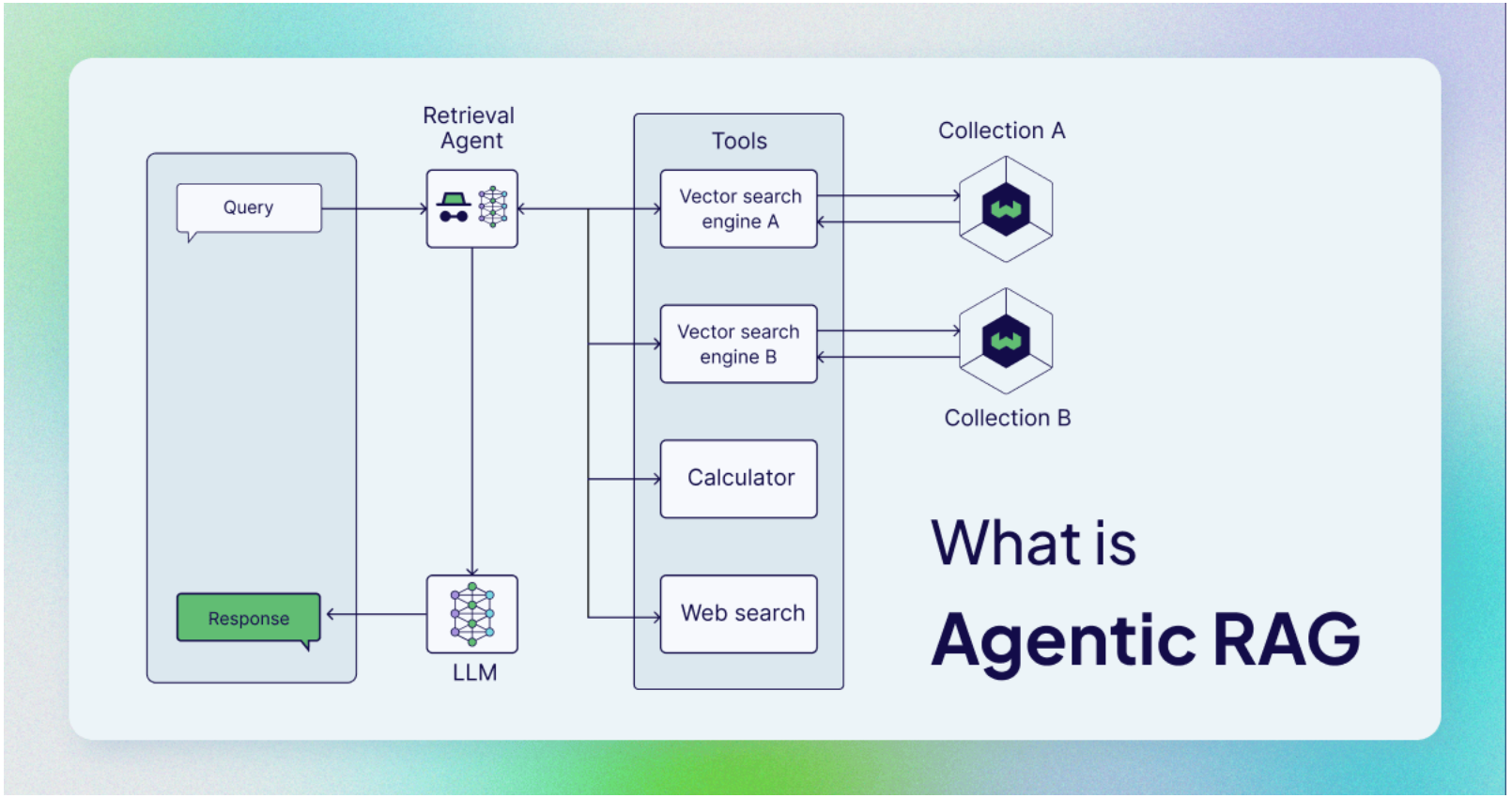
- Authorization on sensitive information
  - Leveled access
- Recency of information
  - How often to update the vector database

- *What may "kill" RAG?*

Agentic RAG

- Traditional RAG: rely on static retrieval and generation
- Agentic RAG: use agents to
  - Decide which data sources to use
  - Formulate queries for information retrieval
  - Integrate external tools like APIs or databases
  - Reason about the user's query and break it down into sub-tasks
  - Iteratively refine their actions and responses

Agentic RAG Pattern



https://weaviate.io/blog/what-is-agentic-rag

Lab 4: RAG for Resumes

- Download the .ipynb file and resumes
- Create a folder “RAG” in your Google Drive
- Unzip and upload resumes folder into “RAG” folder
- There are multiple places that use this path, you need to change them if using your own folder structure

Google NotebookLM

- https://notebooklm.google.com/
- An experimental AI-first notebook that uses RAG technology to integrate user-uploaded documents and notes into its responses, essentially acting as a personalized research assistant

- BloombergGPT Podcast
- https://notebooklm.google.com/notebook/d97d629d-bc27-41df-b154-ab239eaf32c3/audio