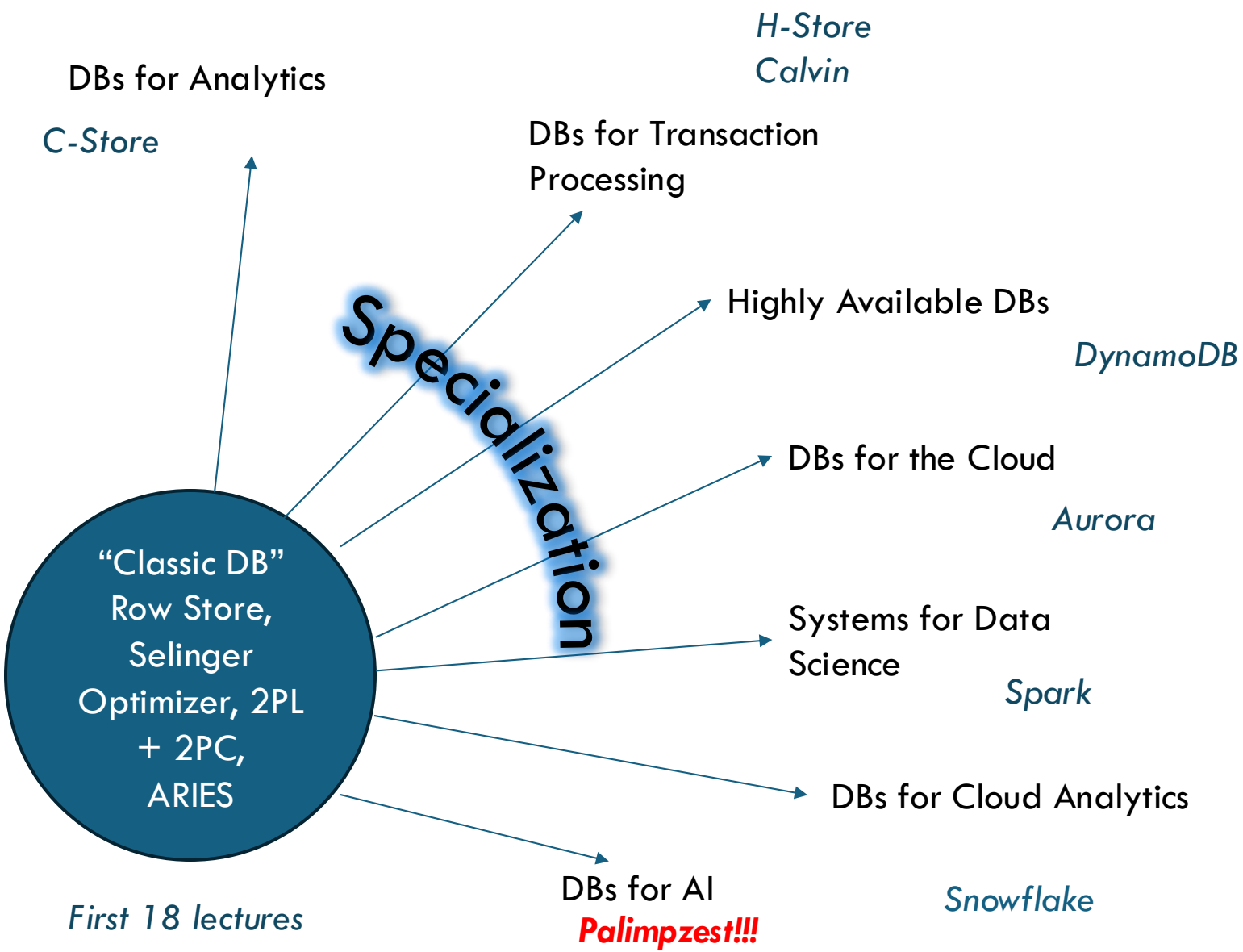
Model of Leonardo's Mechanical Knight, original design from 1495

Databases for Artificial Intelligence

December 4, 2024

Where Are We???


DBs for Analytics
C-Store
H-Store
Calvin
DBs for Transaction Processing
Highly Available DBs
DynamoDB
Specialization
DBs for the Cloud
Aurora
"Classic DB" Row Store, Selinger Optimizer, 2PL + 2PC, ARIES
Systems for Data Science
Spark
DBs for Cloud Analytics
Snowflake
DBs for AI
Palimpzest!!!
First 18 lectures

AI Foundation Models are Full of Promise

- Chat is fun, but foundation models are incredible potential building blocks for apps that fluidly mix AI and data processing

And Are Still Underexploited

- Chat is fun, but foundation models are incredible potential building blocks for apps that fluidly mix AI and data processing

Data Integration	**Multimodal Document Compliance**
Data Cleaning	**Next-Generation Dashboards**
Information Extraction	**Log-Driven System Diagnosis**
Long Document Understanding	**Data-Driven Digital Twins**
Multimodal Scientific Discovery	**… and many others**

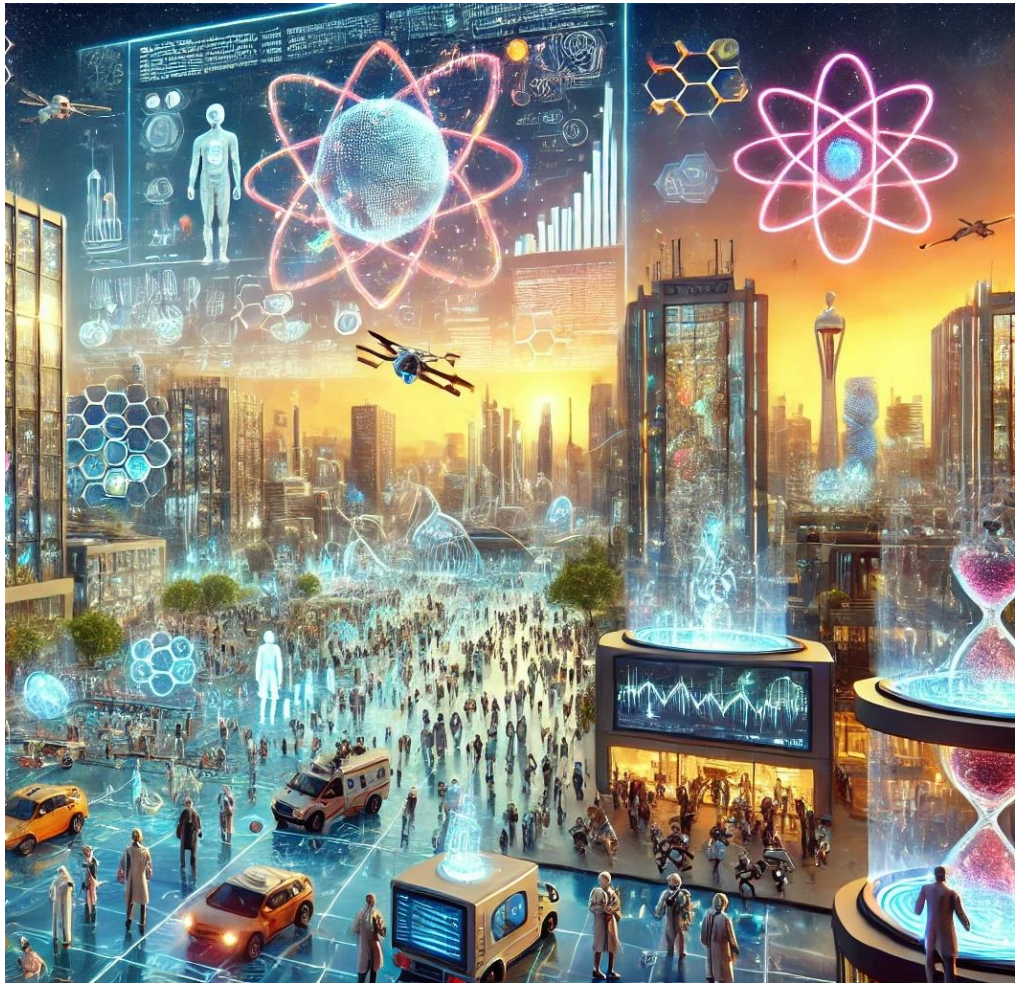
- All of these have traditionally been very difficult to engineer

AI+Data Programs Can Be Thrilling…

Scientific Discovery: ***"Find all the materials science papers that talk about EV batteries"***

Multimodal Document Processing: ***"Double-check all the facts in this mortgage application"***

Effective Government***: "Find all US banks' SEC filings in 2022 and extract footnotes that talk about solvency"***



…but AI Programming is a Drag

Make it fast, cheap, and high quality

…but AI Programming is a Drag

Make it fast, cheap, and high quality

While models, GPUs, and AI methods change every day

…but AI Programming is a Drag

Make it fast, cheap, and high quality

While models, GPUs, and AI methods change every day

While project needs change over time

…but AI Programming is a Drag

Make it fast, cheap, and high quality

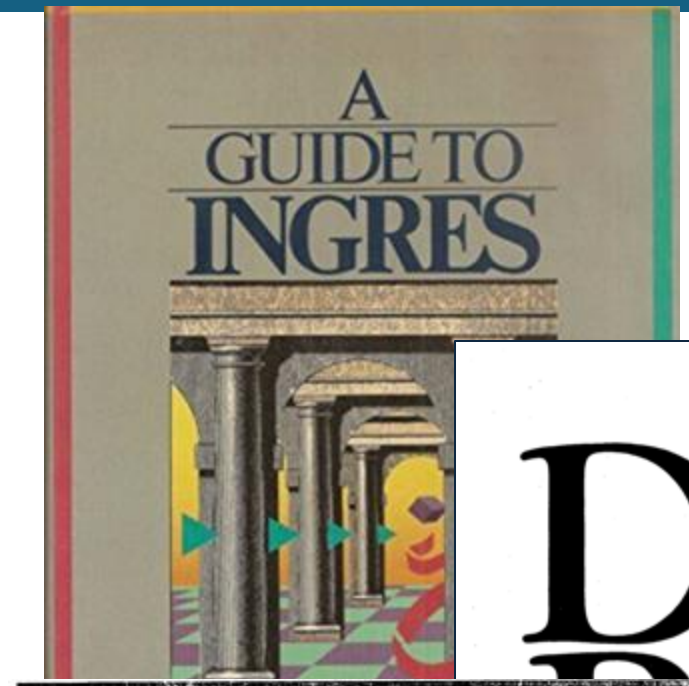
While models, GPUs, and AI methods change every day

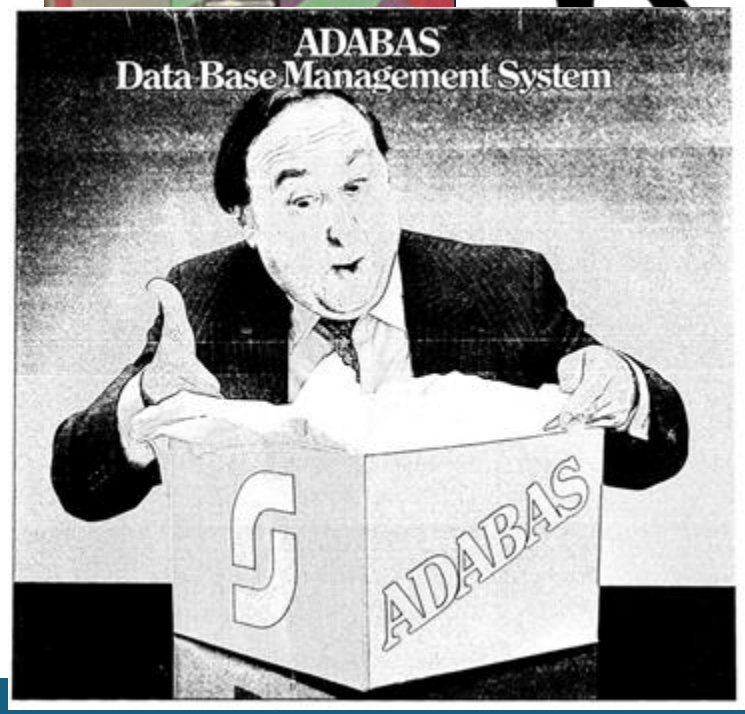
While project needs change over time

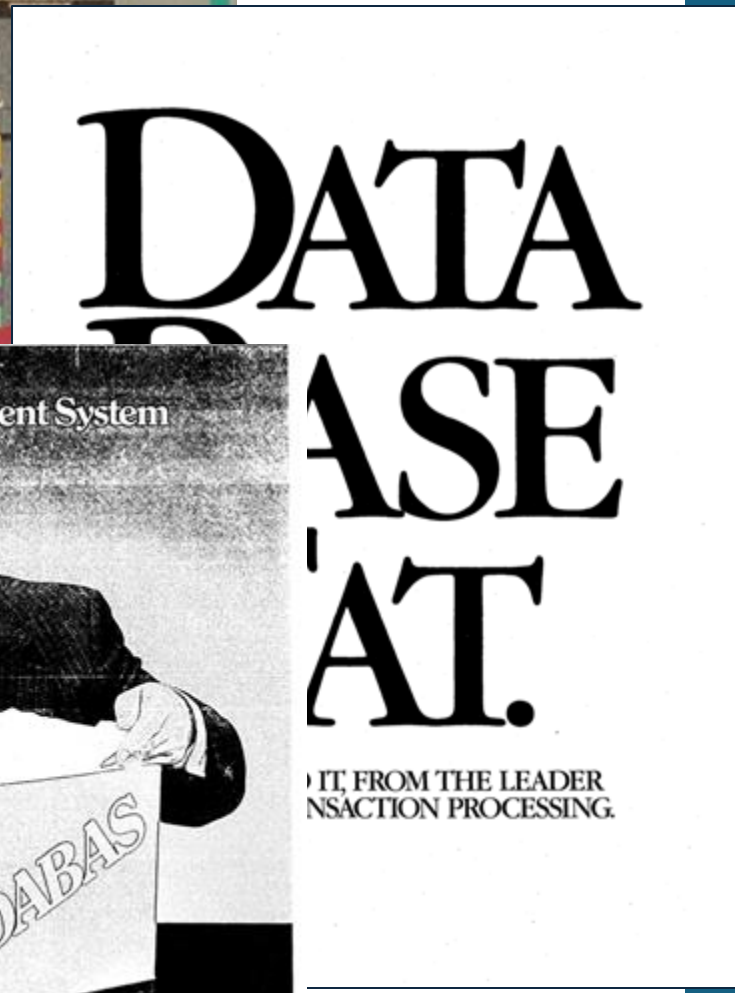
And keep spending flat (at least predictable)

The Good News

- We've solved a problem like this before!
- In the mid-1970s, database programmers had to write custom code for every query
- ***Declarative queries*** allowed them to write succinct programs while also obtaining good performance in a rapidly-changing technological environment
- Let's do the same for AI applications







Our System: Palimpzest

- Python package that lets users implement AI tasks in little code
- Behind the scenes, it hypothesizes and tests 1000s of ways to use AI models to implement user's goal
- It chooses the fastest, cheapest, highest-quality option. When models or prices or hardware change, it will choose differently

["Palimpzest: Optimizing AI-Powered Analytics with Declarative Query Processing", CIDR 25]

Sample AI Application: Real Estate Search

Collect real estate listings; images and text

Home List Price
$1,550,000

Property Highlights
Home Type
Condominium

Parking
Attached, Off Street
...

About 161 Auburn St Unit 161
Built in 2015, this 1763 sq ft contemporary townhouse is only minutes away from the heart of Central Square…

Sample AI Application: Real Estate Search

Collect real estate listings; images and text

Make sure the listing is in my price range

Make sure the listing is within 2 miles of MIT

Make sure it is “modern and attractive” and ”has natural sunlight”

Home List Price
$1,550,000

Property Highlights
Home Type
Condominium

Parking
Attached, Off Street
...

About 161 Auburn St Unit 161 Built in 2015, this 1763 sq ft contemporary townhouse is only minutes away from the heart of Central Square…

Sample AI Application: Real Estate Search

Collect real estate listings; images and text

Make sure the listing is in my price range

Make sure the listing is within 2 miles of MIT

Make sure it is “modern and attractive” and ”has natural sunlight”

Output the results

Home List Price
$1,550,000

Property Highlights
Home Type
Condominium

Parking
Attached, Off Street
...

About 161 Auburn St Unit 161
Built in 2015, this 1763 sq ft contemporary townhouse is only minutes away from the heart of Central Square…

Demo AI Application: Multimodal Real Estate Search

```
# Core PZ code
listings = pz.Dataset("real-estate-tiny", schema=RealEstateListingFiles)
listings = listings.convert(TextRealEstateListing, depends_on="text_content")
listings = listings.convert(ImageRealEstateListing, image_conversion=True,
       depends_on="image_contents")
listings = listings.filterByStr(
"The interior is modern and attractive, and has lots of natural sunlight",
depends_on=["is_modern_and_attractive", "has_natural_sunlight"]
)

listings = listings.filterByFn(within_two_miles_of_mit, depends_on="address")
listings = listings.filterByFn(in_price_range, depends_on="price")

policy = pz.MaxQuality()

results, plan = pz.Execute(listings, policy, num_samples=2,
          nocache=True, verbose=True)
```

Demo AI Application: Multimodal Real Estate Search

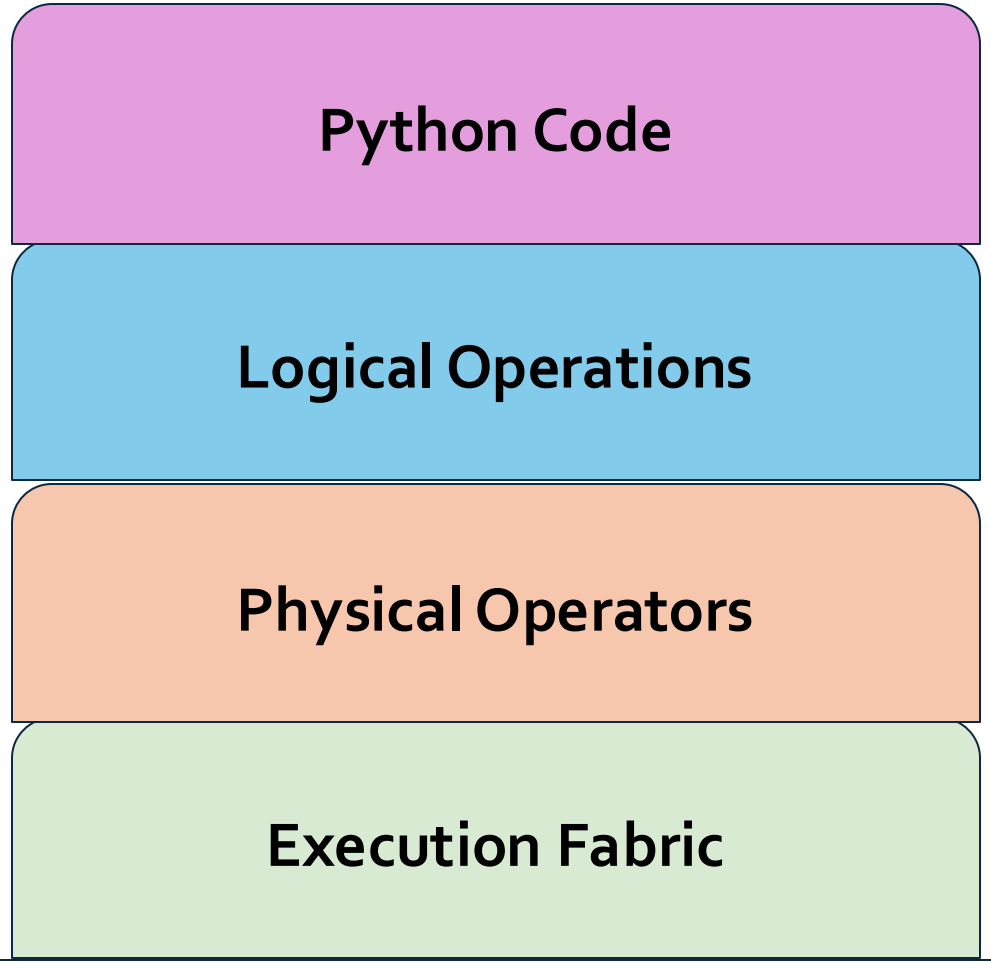
```
# Core PZ code
listings = pz.Dataset("real-estate-tiny", schema=RealEstateListingFiles)
listings = listings.convert(TextRealEstateListing, depends_on="text_content")
listings = listing                                          nversion=True,
       depends_on=
listings = listin
"The interior is                                            ral sunlight",
depends_on=["is_m                                           t"]
)

listings = listin                                           nds_on="address")
listings = listin                                           rice")

policy = pz.MaxQuality()

results, plan = pz.Execute(listings, policy, num_samples=2,
          nocache=True, verbose=True)
```

About 14 lines of interesting code, plus some boilerplate

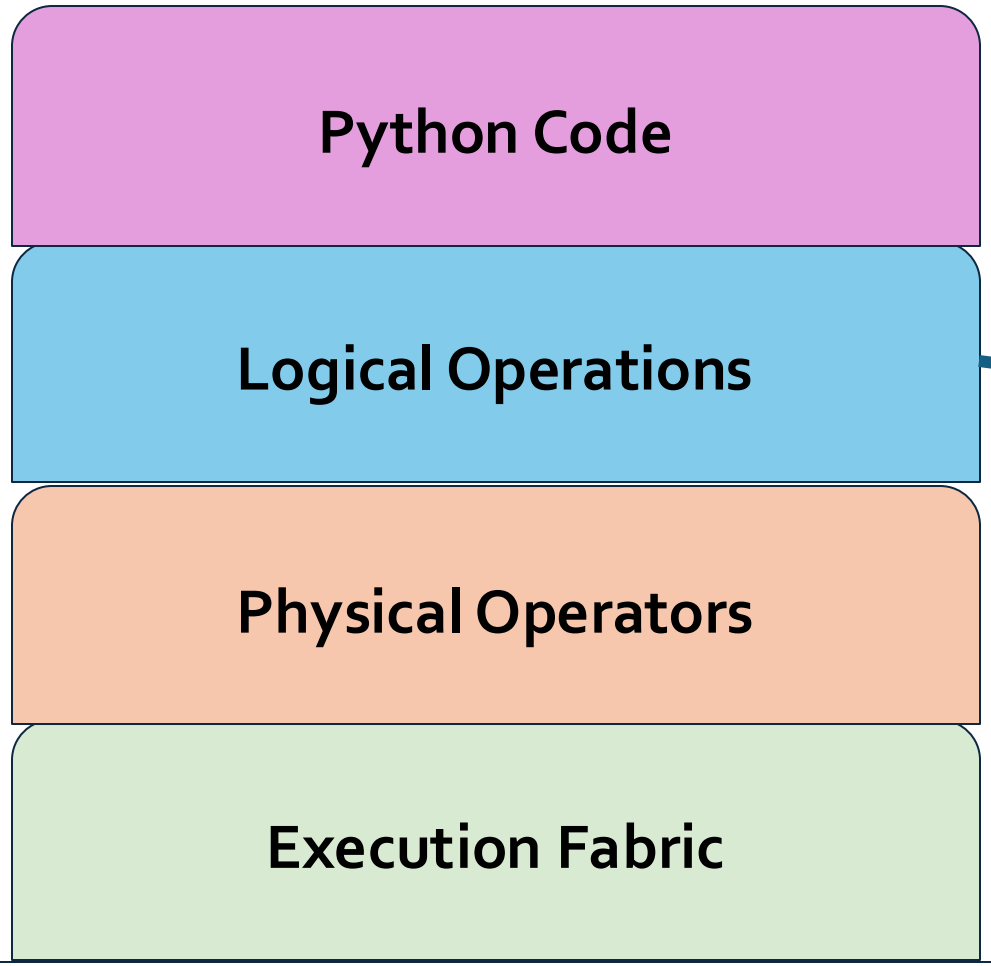
No prompt-writing, data labeling, or profound AI insight needed

Palimpzest Internals

Palimpzest Internals

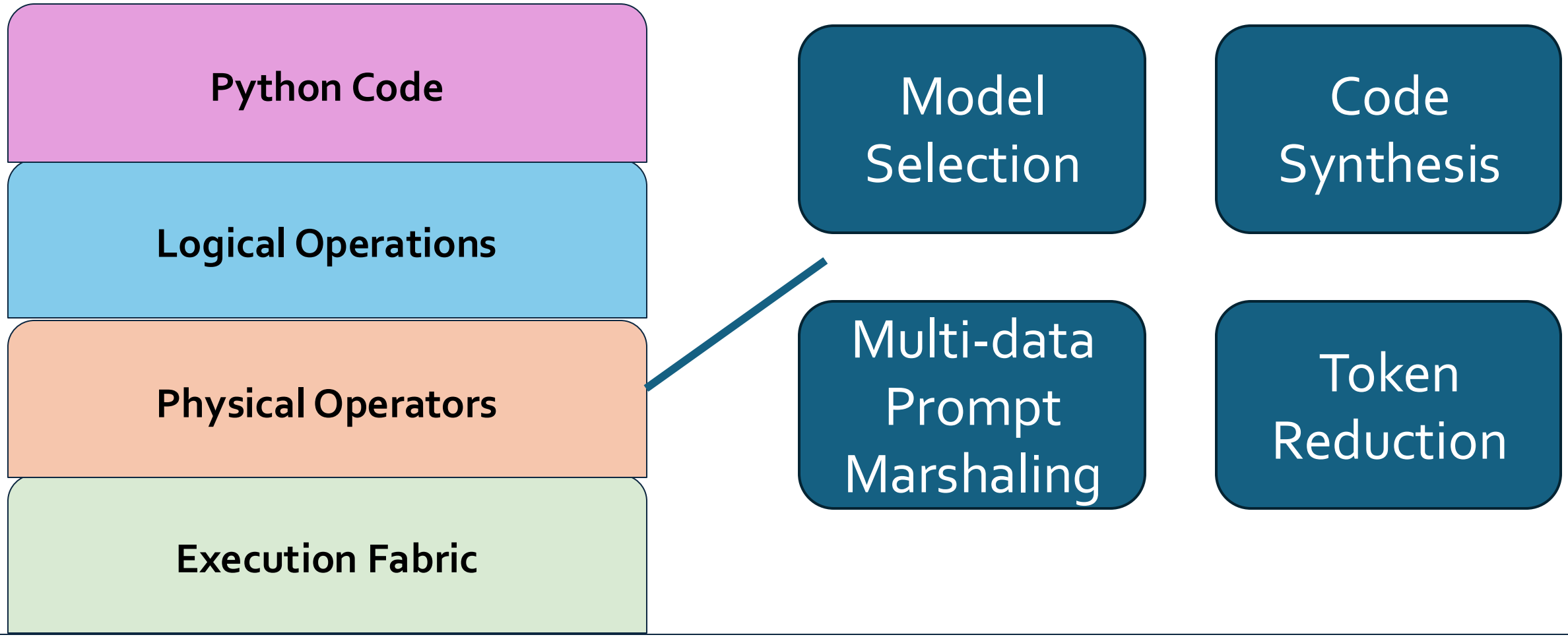


Palimpzest Internals



operator	description
Project	π(rel., cols)
Select	σ(rel., predicate)
Convert	χ(rel., schema_a, schema_b)
Group By	Γ(rel., group_cond., agg.)
Limit	L(rel., limit)
Agg.	α(rel., agg_func)

Palimpzest Internals



Token Reduction: Ideal Pipeline

Paper PDF

Phosphorylation of Exo1 modulates homologous recombination repair of DNA double-strand breaks

[Paper Text]

Phosphorylation of Exo1 modulates homologous recombination repair of DNA double-strand breaks.

Emma Bolderson[1], Nozomi Tomimatsu[2], Derek J. Richard[1], Didier Boucher[1], Rakesh Kumar[3], Tej K. Pandita[3], Sandeep Burma[2] and Kum Kum Khanna[1]* [1]Signal Transduction Laboratory, Queensland Institute of Medical Research, Brisbane, Queensland 4029, Australia, [2]Department of Radiation Oncology, UT Southwestern Medical Center at Dallas, Dallas, TX 75390-9187, [3]Department of Radiation Oncology, Washington University School of Medicine, St. Louis, MO 63108, USA Received October 22, 2009; Revised November 18, 2009; Accepted November 24, 2009

ABSTRACT DNA double-strand break (DSB) repair via the homologous recombination pathway is a multi-stage process, which results in repair of the DSB without loss of genetic information or fidelity. One essential step in this process is the generation of extended single-stranded DNA (ssDNA) regions at the break site. This ssDNA serves to induce cell cycle checkpoints and is required for Rad51 mediated strand invasion of the sister chromatid. Here, we show that human Exonuclease 1 (Exo1) is required for the normal repair of DSBs by HR. Cells depleted of Exo1 show chromosomal instability and hypersensitivity to ionising radiation (IR) exposure. We find that Exo1 accumulates rapidly at DSBs and is required for the recruitment of RPA and Rad51 to sites of DSBs, suggesting a role for Exo1 in ssDNA generation. Interestingly, the phosphorylation of Exo1 by ATM appears to regulate the activity of Exo1 following resection, allowing optimal Rad51 loading and the completion of HR repair. These data establish a role for Exo1 in resection of DSBs in human cells, highlighting the critical requirement of Exo1 for DSB repair via HR and thus the maintenance of genomic stability.

INTRODUCTION DNA double-strand breaks (DSBs) can be induced by a variety of factors such as chemotherapeutic agents, ionizing radiation (IR), or the collapse of cellular metabolism. These breaks trigger a complex network of signaling pathways involved in the detection…

LLM Prompt

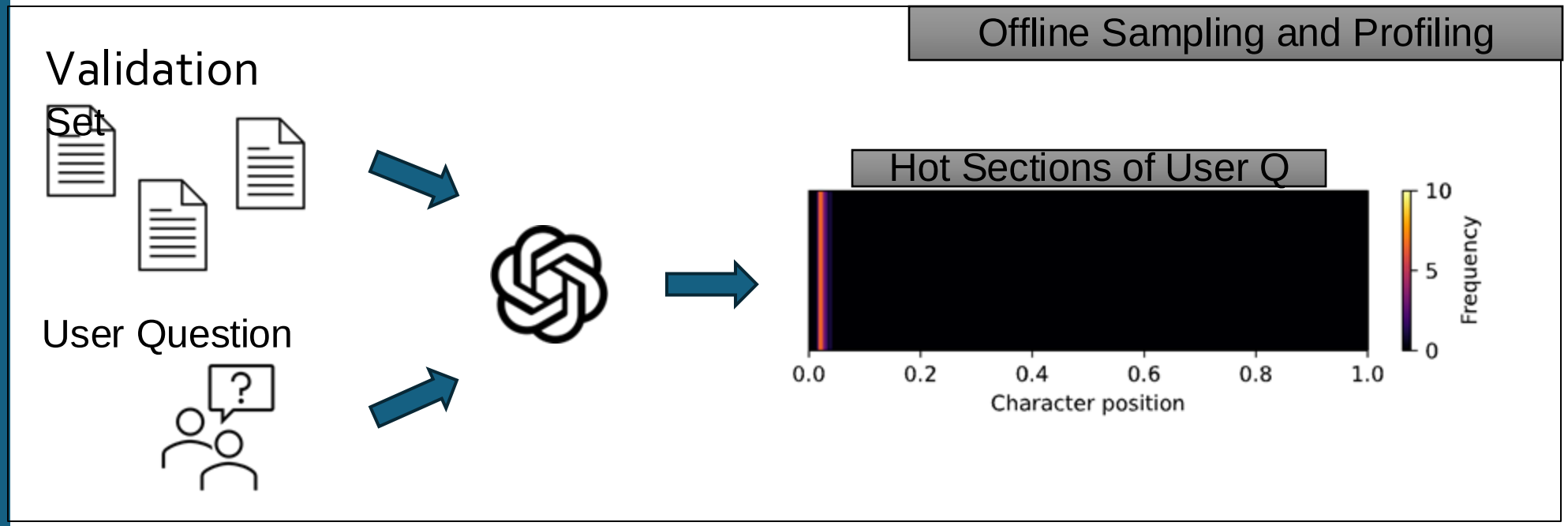
Context : [highlighted_text]
Question: What are the authors of this paper?

LLM Answer

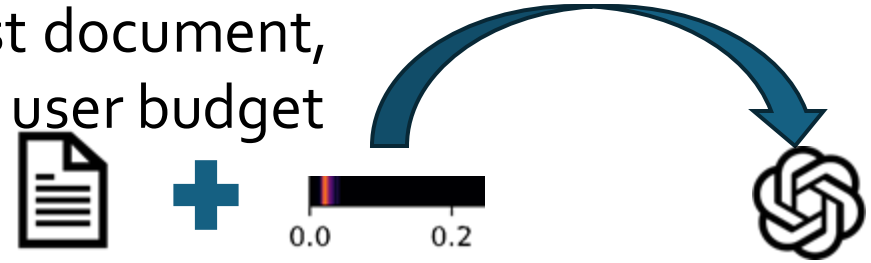
Bolderson, Tomimatsu, Richard, Boucher, Kumar, …

The paper text varies in its relevance to a given query. How can we find the relevant text chunks beforehand?

Token Reduction Workflow

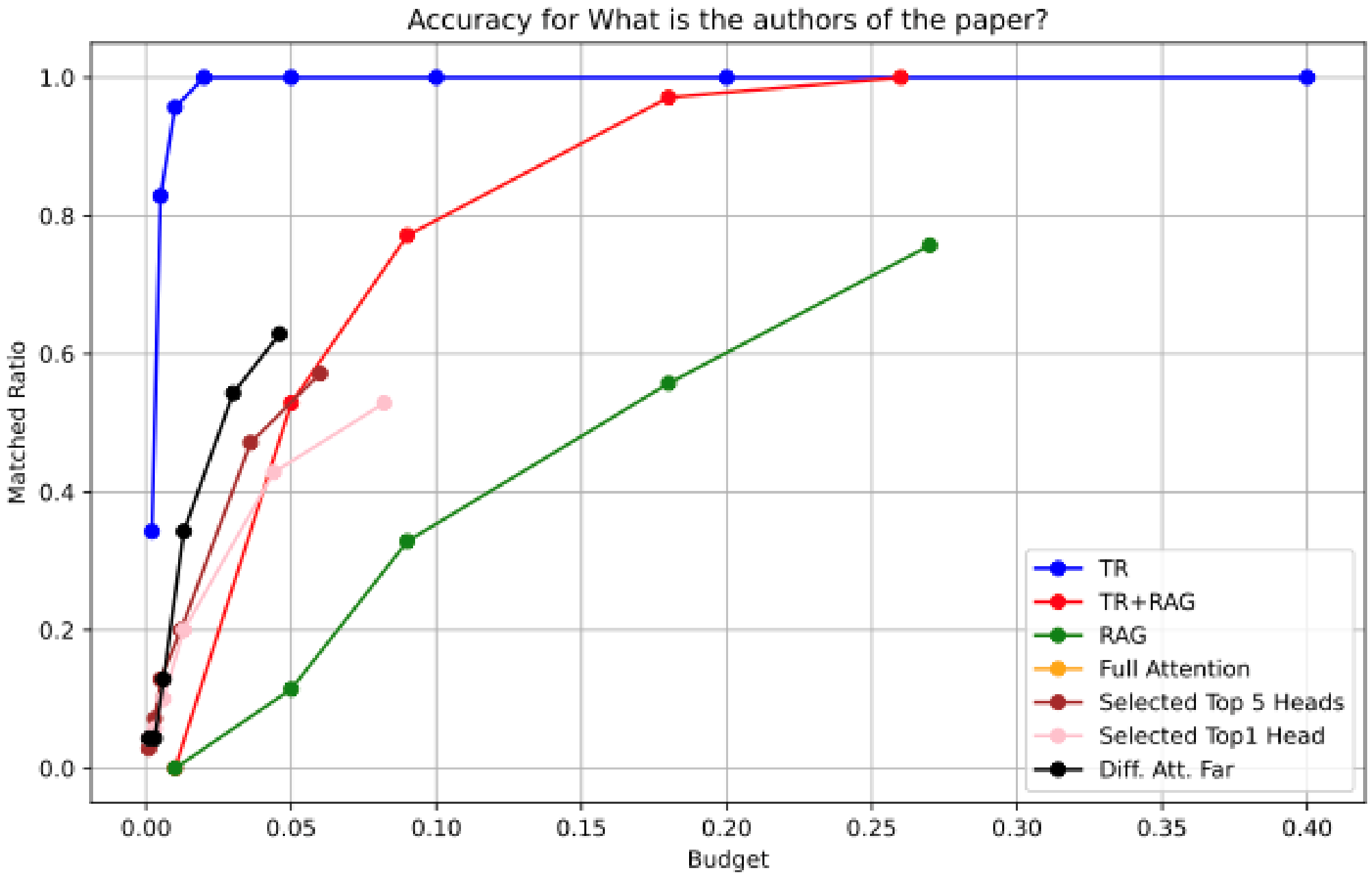


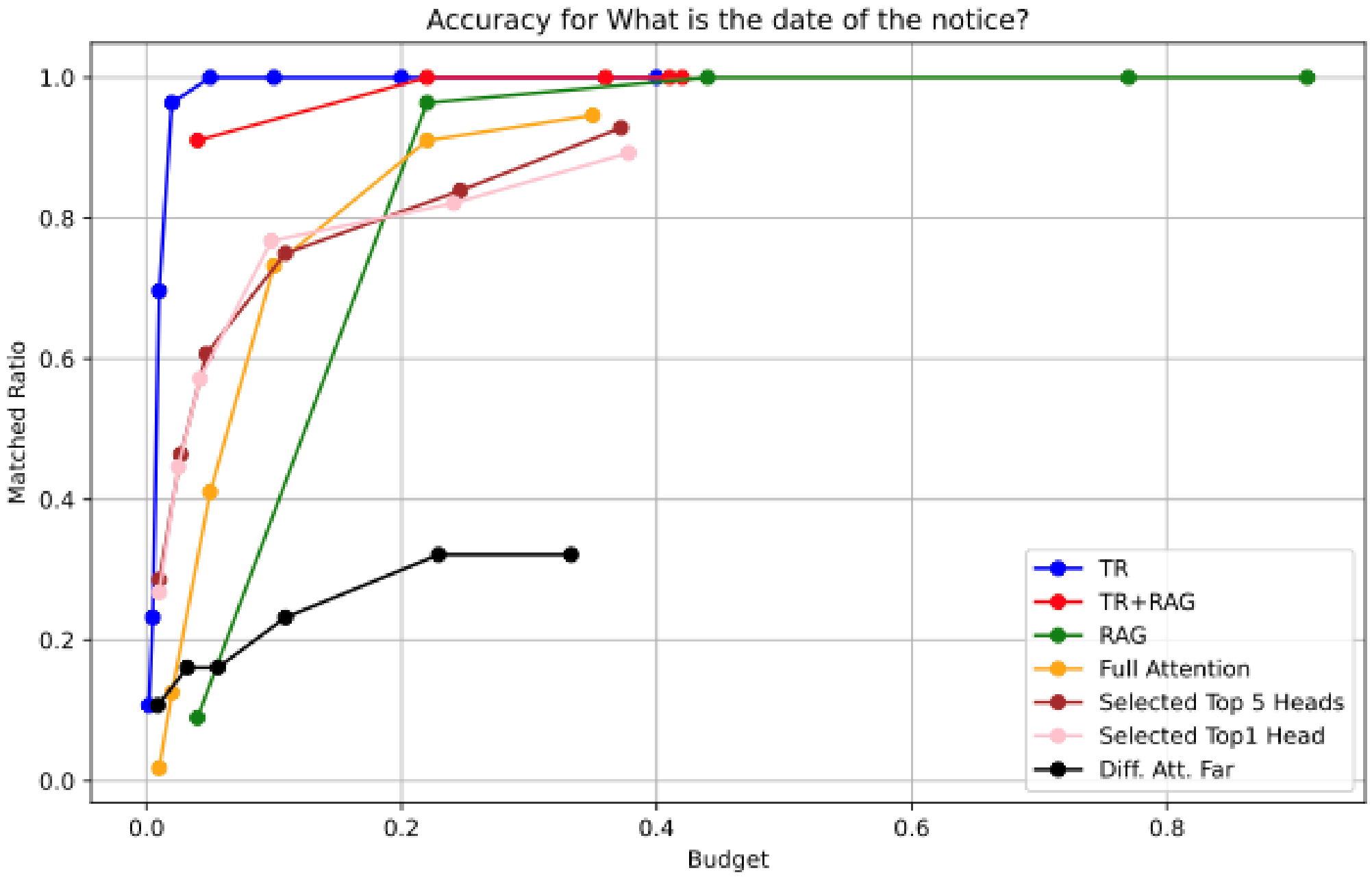
Input: test document,
max user budget

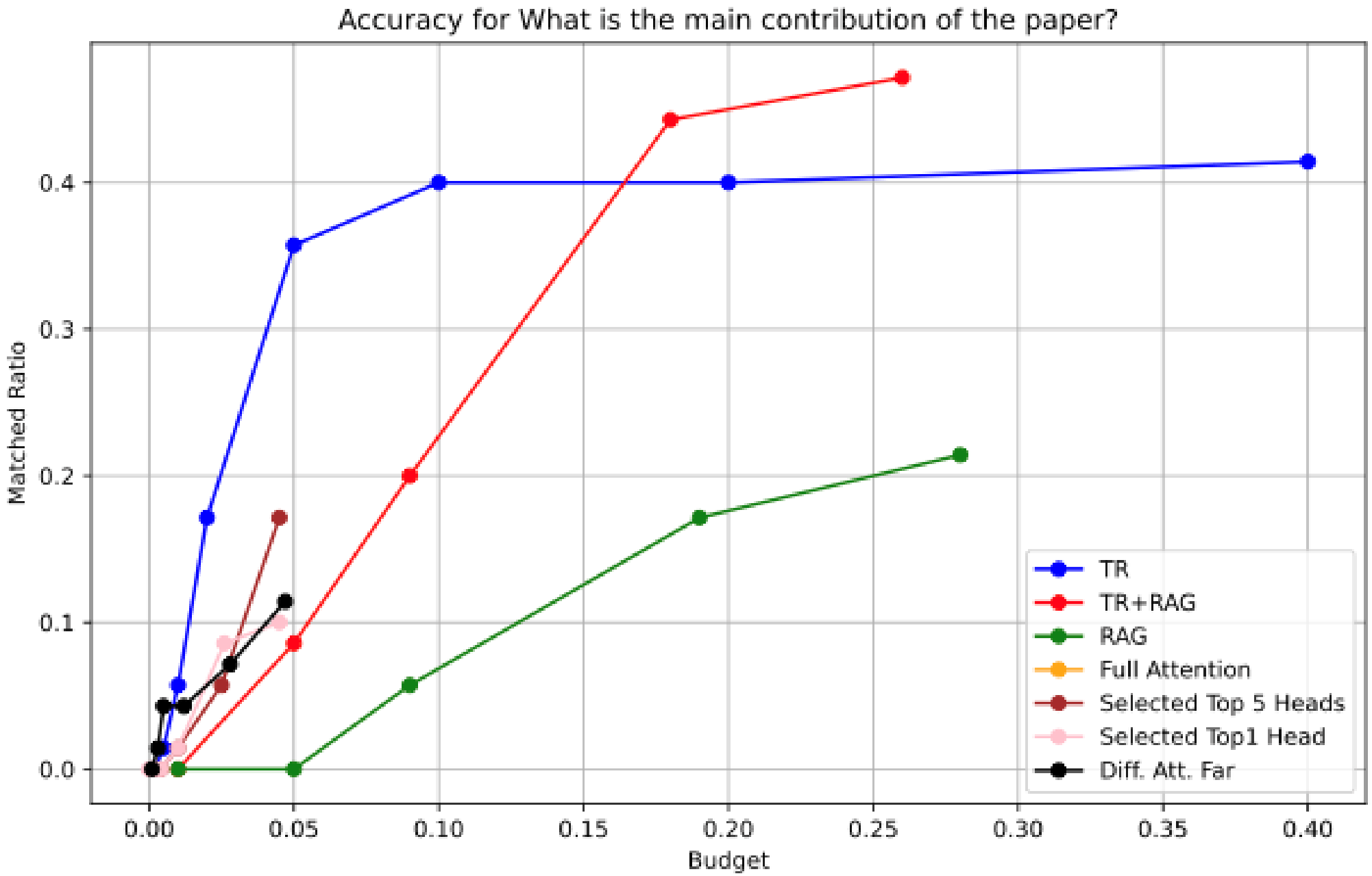


Optimal range:

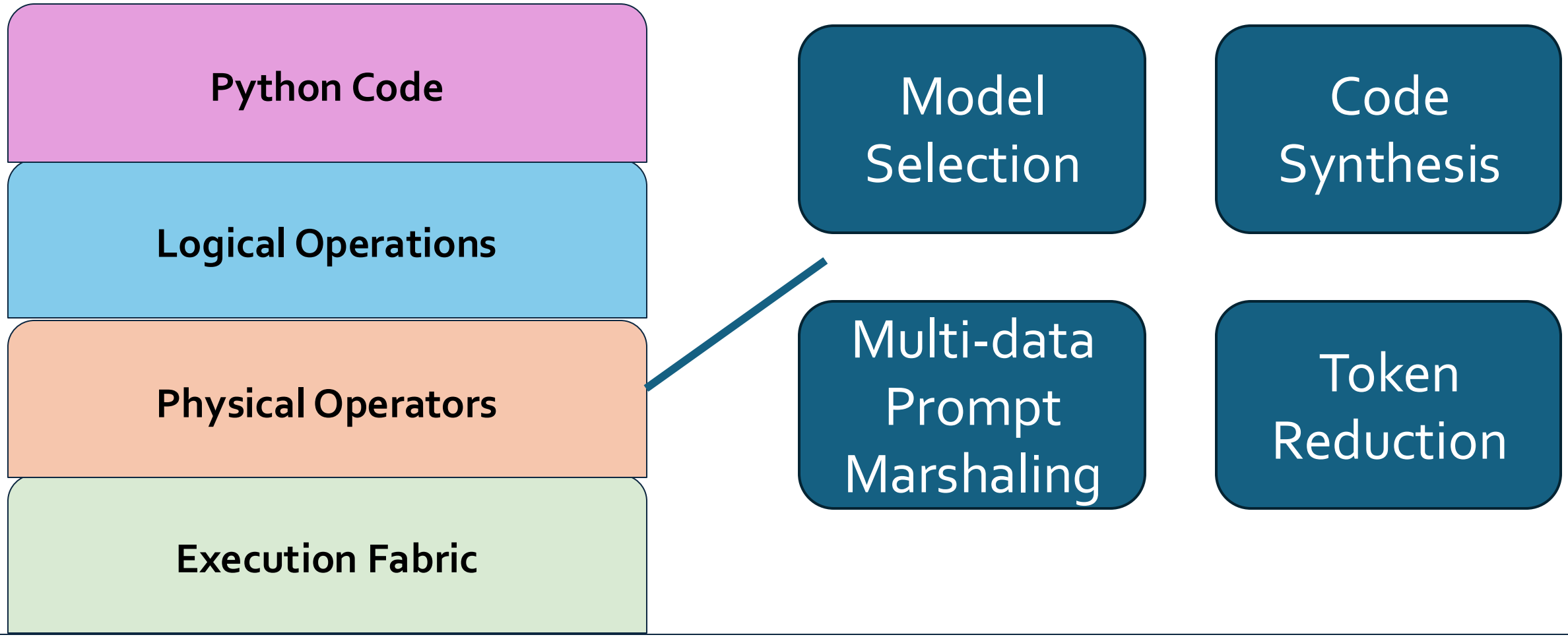
1. Comparable results
2. Lower $ cost
3. Lower runtime
4. Memory saving


Accuracy for What is the authors of the paper?
Matched Ratio
Budget
1.0
0.8
0.6
0.4
0.2
0.0
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
TR
TR+RAG
RAG
Full Attention
Selected Top 5 Heads
Selected Top1 Head
Diff. Att. Far

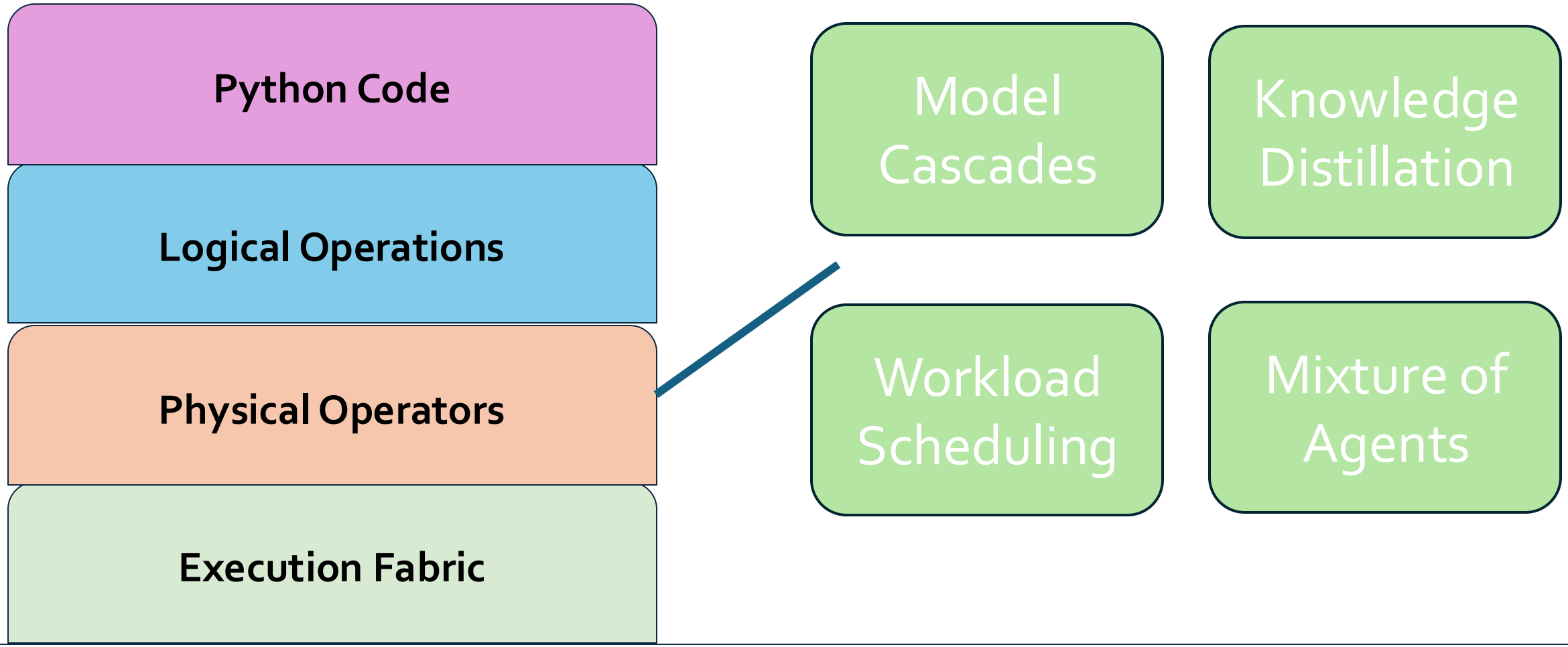

Accuracy for What is the date of the notice?
Matched Ratio
Budget
0.0
0.2
0.4
0.6
0.8
1.0
TR
TR+RAG
RAG
Full Attention
Selected Top 5 Heads
Selected Top1 Head
Diff. Att. Far


Accuracy for What is the main contribution of the paper?
Matched Ratio
0.0
0.1
0.2
0.3
0.4
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
Budget
TR
TR+RAG
RAG
Full Attention
Selected Top 5 Heads
Selected Top1 Head
Diff. Att. Far

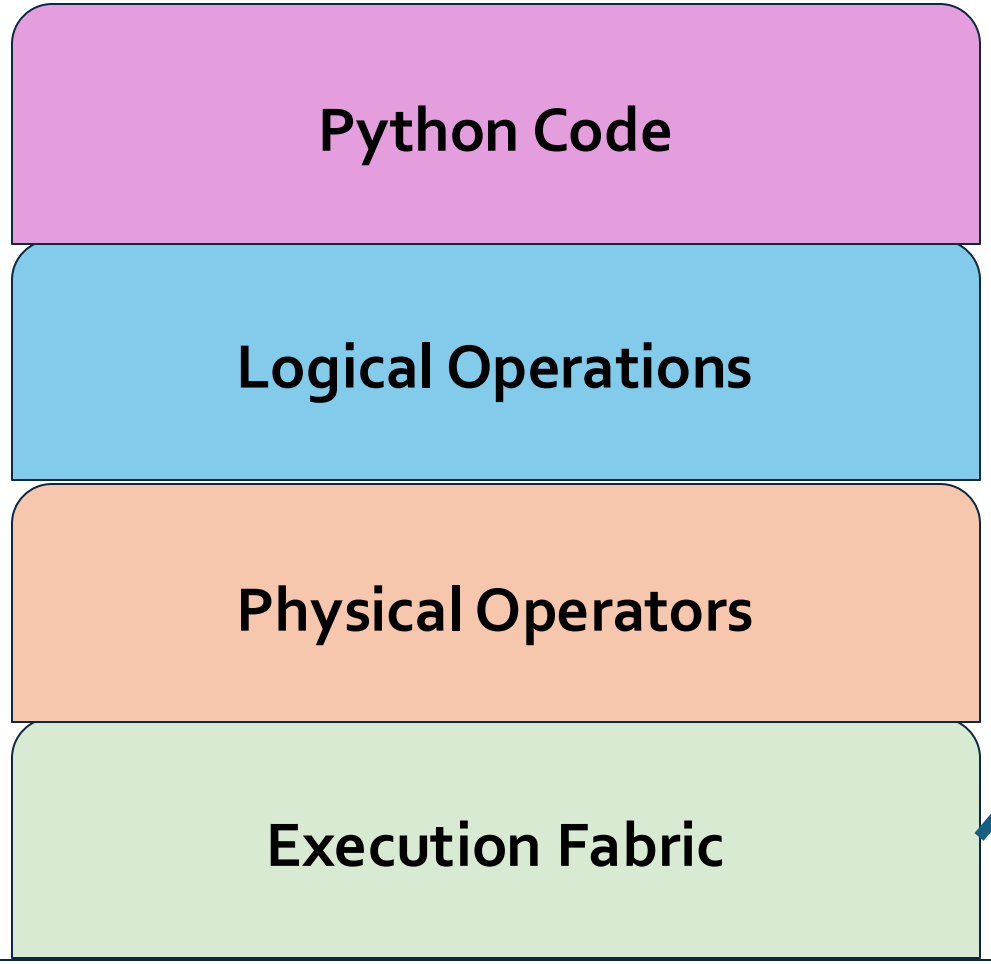
Palimpzest Internals



Palimpzest Internals



Palimpzest Internals



The PZ optimizer enumerates physical plans, estimates quality, runtime, cost for each

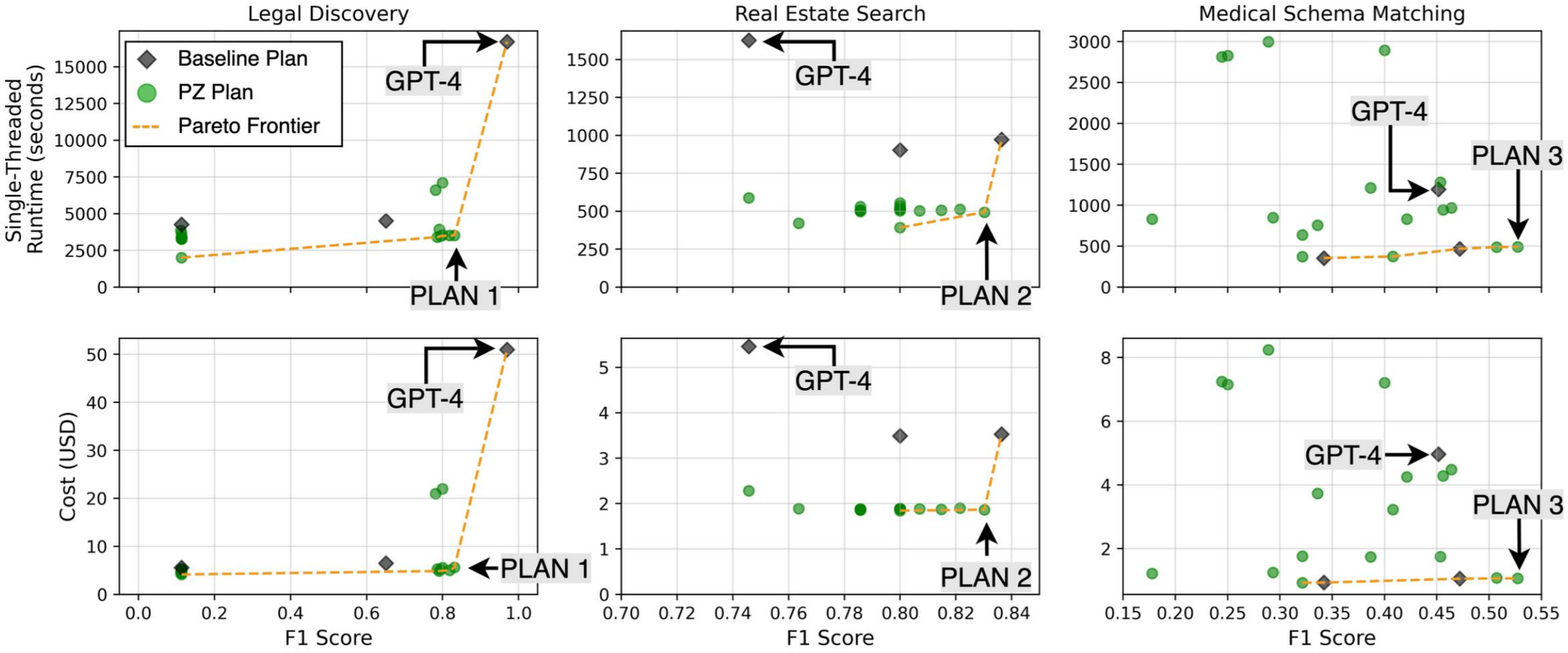
It picks the plan that best matches the user's desired tradeoffs

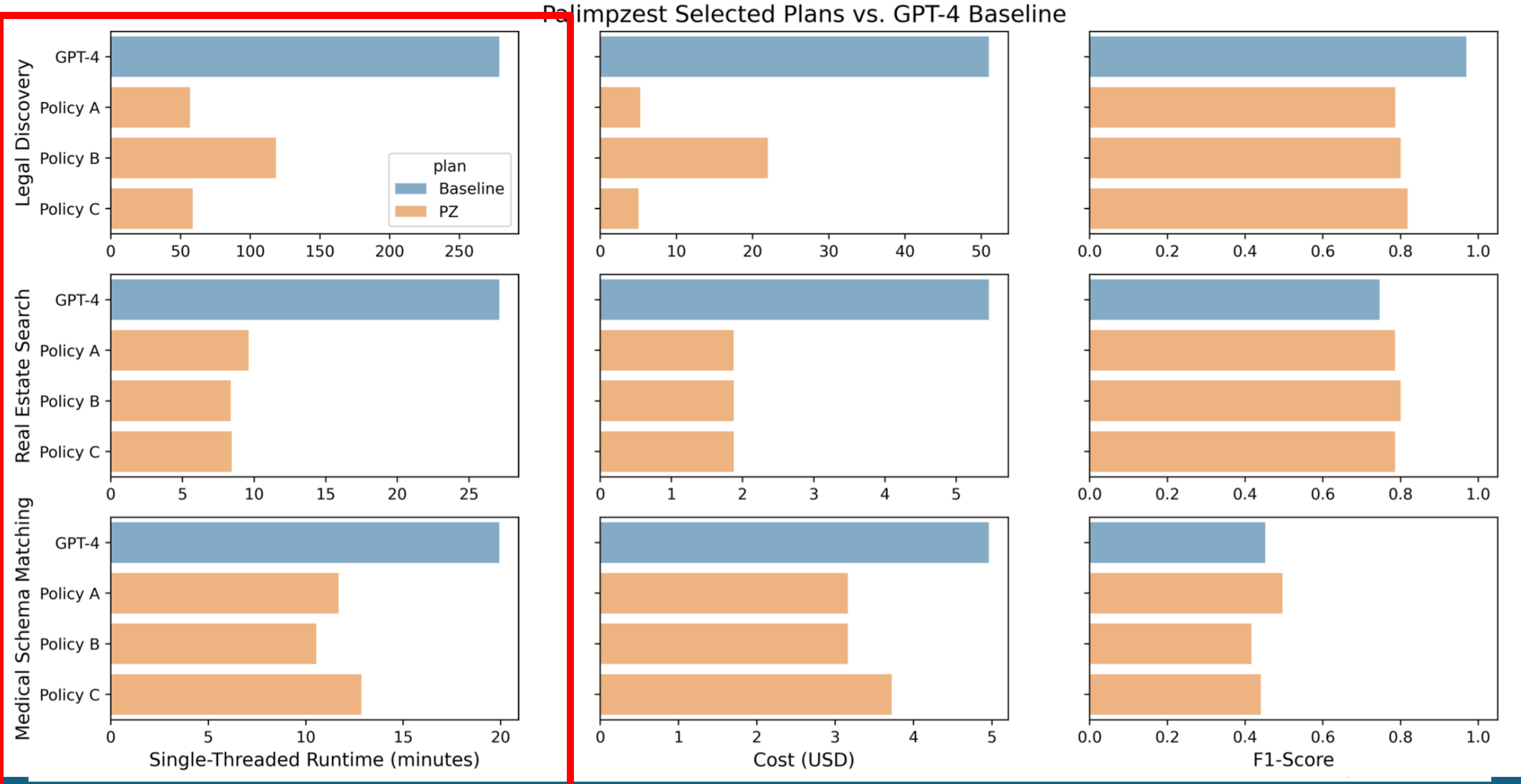
Estimating quality is the hardest part. Current implementation uses a "champion model"

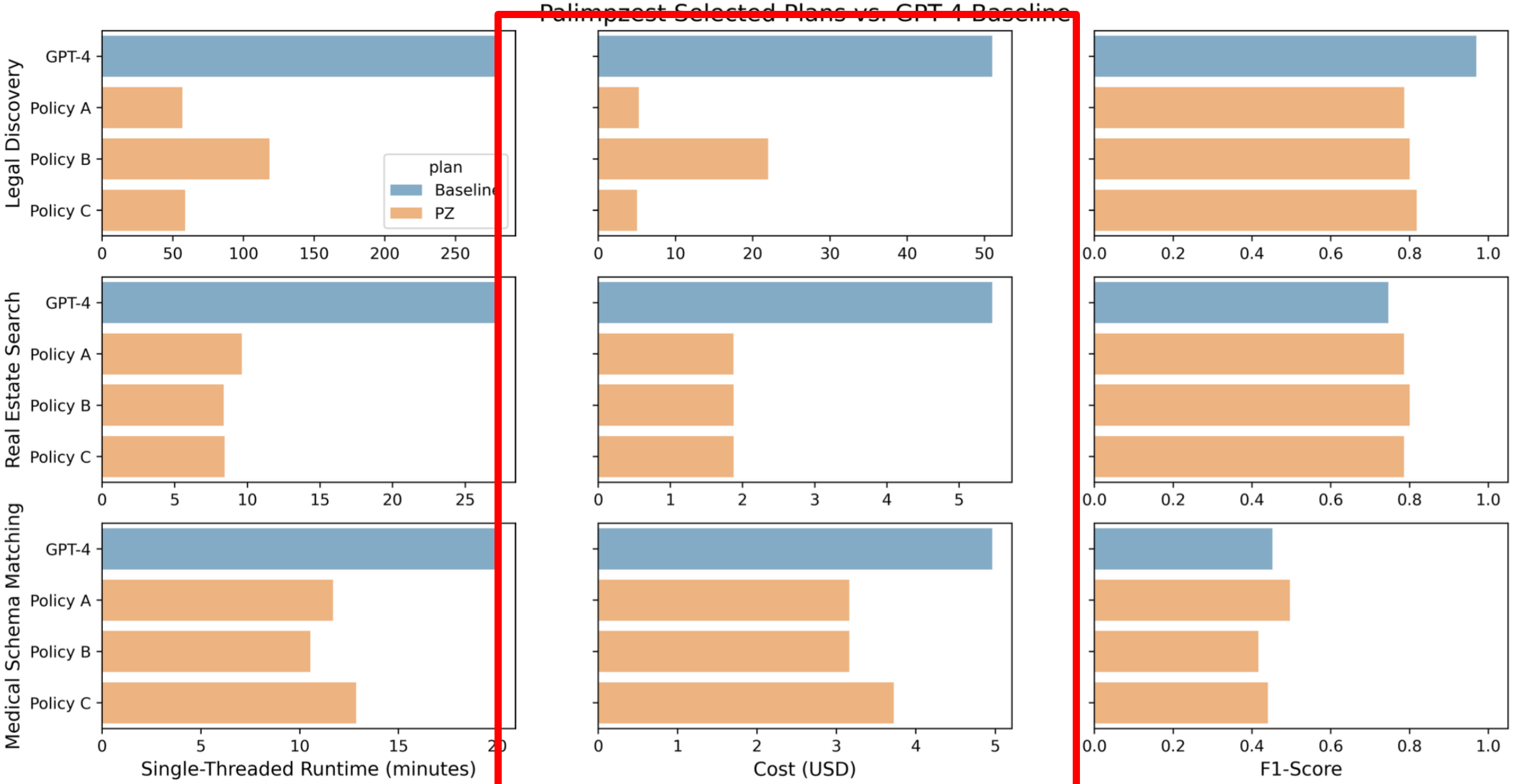
Prototype and Experiments

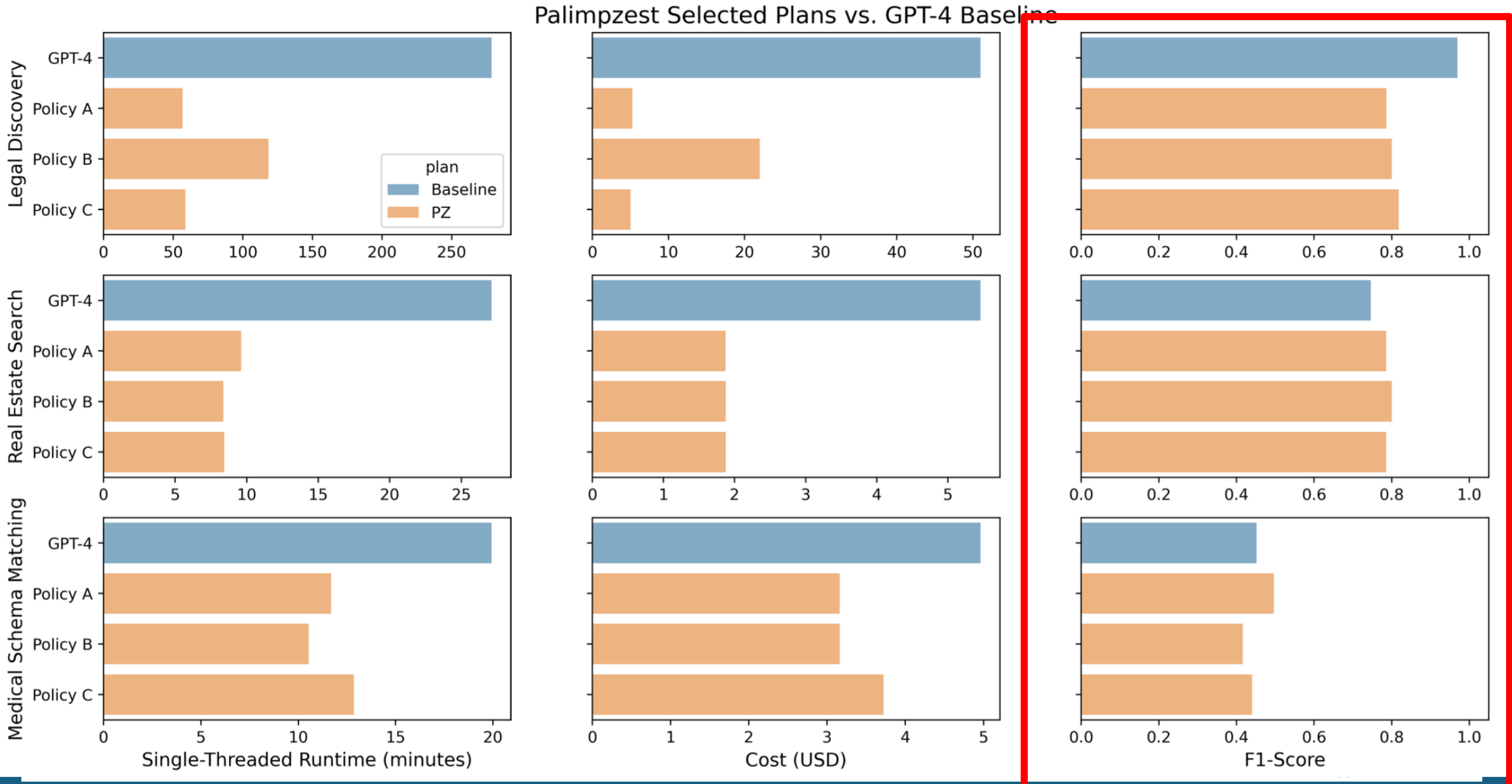
- Implemented in about 9800 lines of Python
- Claims:
  - Physical optimizations can produce better plans than a naïve program would obtain
  - The optimizer can successfully identify these plans
- Workloads:
  - Multimodal Real Estate Search (above task; 100 listings, both text and images; 14 LOC)
  - Legal Discovery (identify fraudulent intent; 1000 emails; 17 LOC)
  - Medical Schema Matching (reproduce a real-world data integration task for cancer researchers; 11 spreadsheets with 49 tables; 30 LOC)

Good Physical Plans Exist




Palimpzest Selected Plans vs. GPT-4 Baseline
Legal Discovery
Real Estate Search
Medical Schema Matching
GPT-4
Policy A
Policy B
Policy C
plan
Baseline
PZ
0
50
100
150
200
250
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
0
1
2
3
4
5
0
5
10
15
20
Single-Threaded Runtime (minutes)
Cost (USD)
F1-Score


Palimpzest Selected Plans vs. GPT-4 Baseline
Legal Discovery
Real Estate Search
Medical Schema Matching
GPT-4
Policy A
Policy B
Policy C
plan
Baseline
PZ
Single-Threaded Runtime (minutes)
Cost (USD)
F1-Score


Palimpzest Selected Plans vs. GPT-4 Baseline
Legal Discovery
Real Estate Search
Medical Schema Matching
GPT-4
Policy A
Policy B
Policy C
plan
Baseline
PZ
0
50
100
150
200
250
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
0
1
2
3
4
5
Single-Threaded Runtime (minutes)
Cost (USD)
F1-Score

Biomedical Use Case: Literature Search

Example:

1. A researcher is investigating a concept, e.g., "Phosphorylation of Exo1"
2. We filter papers in literature to find relevant mentions
3. We scan paper text to solve paper references
4. System returns the relevant text from referenced papers

Current implementation:
27 lines of user code
61% F1
~12s per document

phosphorylation of exo1

Phosphorylation of Exo1 modulates homologous recombination repair of DNA double-strand breaks

gesting that DSB processing is dispensable for activation of the ATM-dependent signaling pathway.

Exo1 is phosphorylated in response to DSBs

Since the activity of Exo1 has been suggested to be regulated by phosphorylation in budding yeast (35), we next investigated whether the phosphorylation of Exo1

stability distinct from ATM activation. *Cell*, **135**, 85–96.
35. Morin,I., Ngo,H.P., Greenall,A., Zubko,M.K., Morrice,N. and Lydall,D. (2008) Checkpoint-dependent phosphorylation of Exo1 modulates the DNA damage response. *EMBO J.*, **27**, 2400–2410.
36. Matsuoka,S., Ballif,B.A., Smogorzewska,A., McDonald,E.R. III,

Checkpoint-dependent phosphorylation of Exo1 modulates the DNA damage response

phosphorylation. Furthermore, mutation of these Exo1 residues altered the DNA damage response to uncapped telomeres and camptothecin treatment, in a manner that suggests Exo1 phosphorylation inhibits its activity. We propose that Rad53-dependent Exo1 phosphorylation is involved in a negative feedback loop to limit ssDNA accumulation and DNA damage checkpoint activation.

Biomedical Use Case: Data Collection

Current implementation:
30 user lines of code
40% F1
~9s runtime per document

Example:

1. A researcher is surveying all data available from the literature
2. The original papers report on different sources, may contain supplemental data
3. We scan the paper & identify all publicly available data (e.g., through DOI)
4. We automatically crawl and collect the data in a shared repository

a resource for the broader cancer research community to advance cancer diagnosis and treatment.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.ccell.2023.06.009.

ACKNOWLEDGMENTS

The Clinical Proteomic Tumor Analysis Consortium (CPTAC) is supported by the National Cancer Institute of the National Institutes of Health under award

Download: Download Acrobat PDF file (456KB)

Document S1. Data S1 and Figure S1.

Download: Download spreadsheet (510KB)

Table S1. Clinical and demographic information, related to Figure 1.

Download: Download spreadsheet (385KB)

Table S2. Clinical information related to tumor and normal samples, related to Figure 1.

Download: Download text file (26MB)

Table S3. Representative isoforms for each gene in the BCM pipelines, related to Data S1.

Download: Download Acrobat PDF file (2MB)

Document S2. Article plus supplemental information.

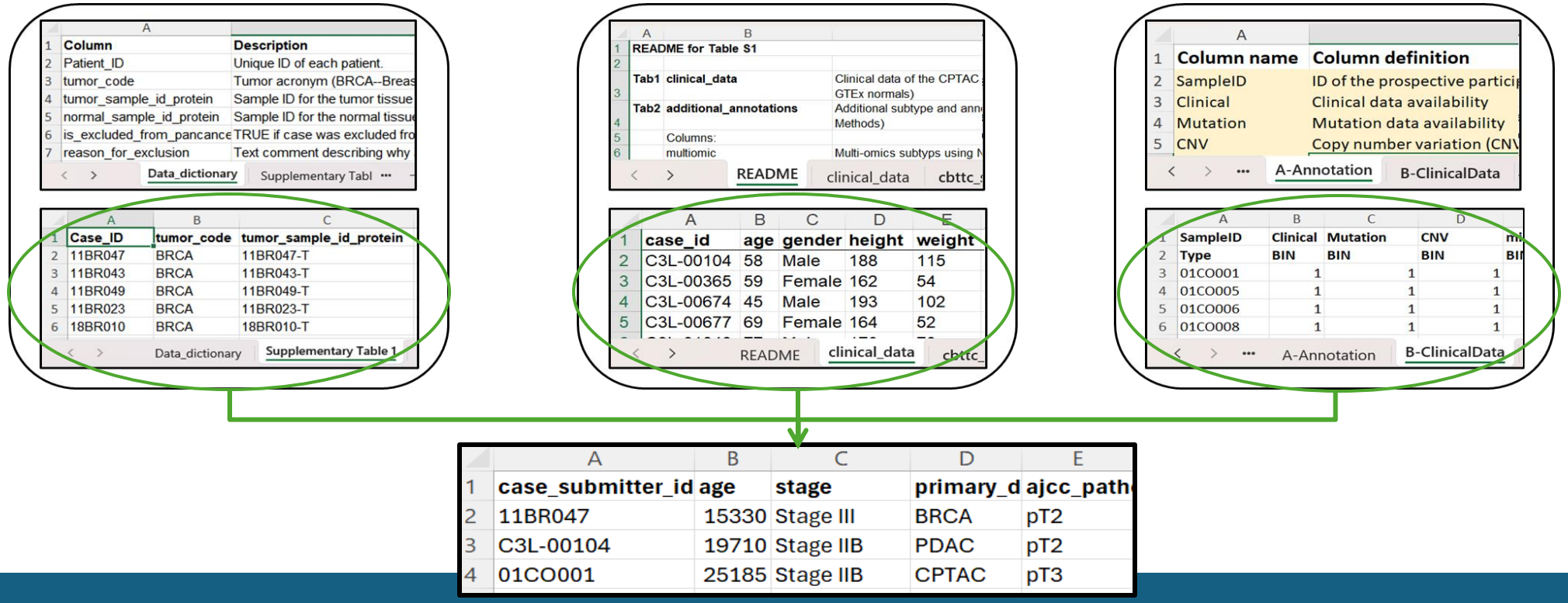
mmc1.xlsx
mmc2.xlsx
mmc3.xlsx
mmc4.xlsx
mmc5.xlsx
mmc6.xlsx
mmc7.xlsx

Biomedical Use Case: Data Harmonization

Example:

1. A researcher wants to run a longitudinal study from several sources
2. The original datasets contain relevant as well as irrelevant tables
3. First, system automatically identifies relevant data
4. Then, system merges data across sources matching columns and values

Current implementation:
35 user lines of code
46% F1
~26s runtime per table



Other Recent Systems

- Many existing RDBMSes (BigQuery, Databricks, Redshift) offer LLM UDFs
- Basic programming and optimization frameworks like LangChain, DSPy are programmer-focused. They don't offer a complete general-purpose query model
- Some systems are focused on information extraction: ZenDB, EVAPORATE
- LOTUS is a general-purpose system similar to Palimpzest. The query language is dataframe-focused, has a different set of optimizations
- DocETL offers a query language tailored for large heterogeneous document collections. Very different set of operators compared to PZ and LOTUS

Hi! I'm your Beaker Agent and I can help you do programming and software engineering tasks.

Feel free to ask me about whatever the context specializes in..

On top of answering questions, I can actually run code in a python environment, and evaluate the results. This lets me do some pretty awesome things like: web scraping, or plotting and exploring data. Just shoot me a message when you're ready to get started.

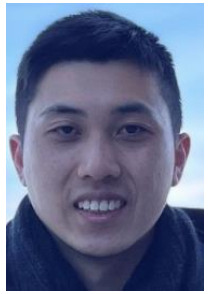
How can the agent help?

The People Who Actually Did The Work

Chunwei Liu

Matt Russo

Zui Chen



Peter Baille Chen

Rana Shahout

Gerardo Vitagliano

Sylvia Zhang

Palimpzest is Basis of Many New Projects

Website: https://dsg.csail.mit.edu/projects/palimpzest/

Paper: https://arxiv.org/pdf/2405.14696

Demo: https://bit.ly/4c6vIcQ

Code: https://github.com/mitdbg/palimpzest

Tim Kraska

Sam Madden

Lei Cao

Mike Franklin