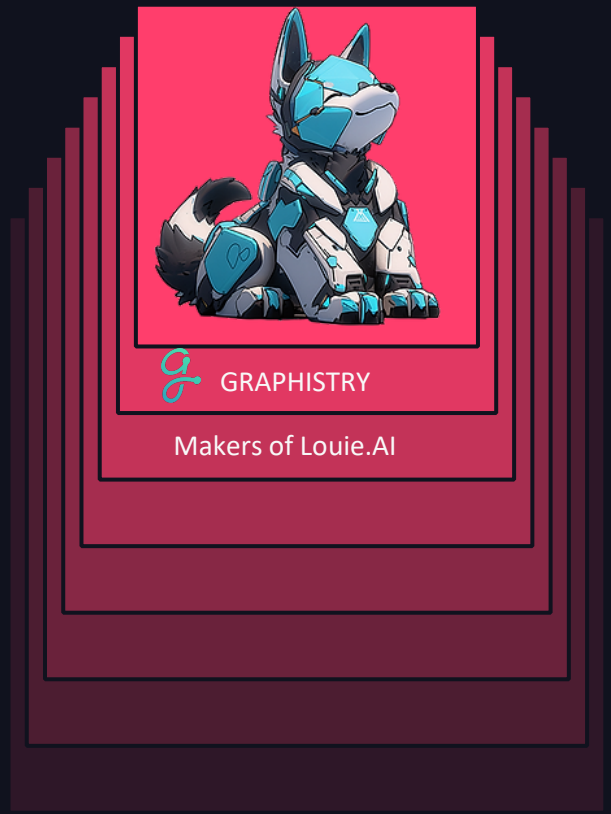
DATA+AI SUMMIT BY databricks

Winning the U.S. Cyber Command AI Alert Data Challenge with Graphs



Dr Alex Morrise
June 13, 2024

GRAPHISTRY

100X Investigations

- ✓ Connect, Use, Embed: Splunk, Databricks, Neo4j, Python, …
- ✓ First GPU visual graph AI platform
- ✓ Louie.AI: GenAI-first investigation & automation

Users Operators/analysts, data scientists, & developers: Cybersecurity, fraud, supply chain, IT, fintech, & more

Distros SaaS, private cloud, air-gapped

The U.S.CYBERCOMM AI Challenge: Imagine being a detective with the power to see crimes. That's what we do with cyber logs.

USCYBERCOM AI Challenge: Landscape of Threats from Fusion Center data

Many bad actors, many locations, and diverse threat models

CESNET: Universities + ISPs

IDS, honeypots, malware, phishing, …

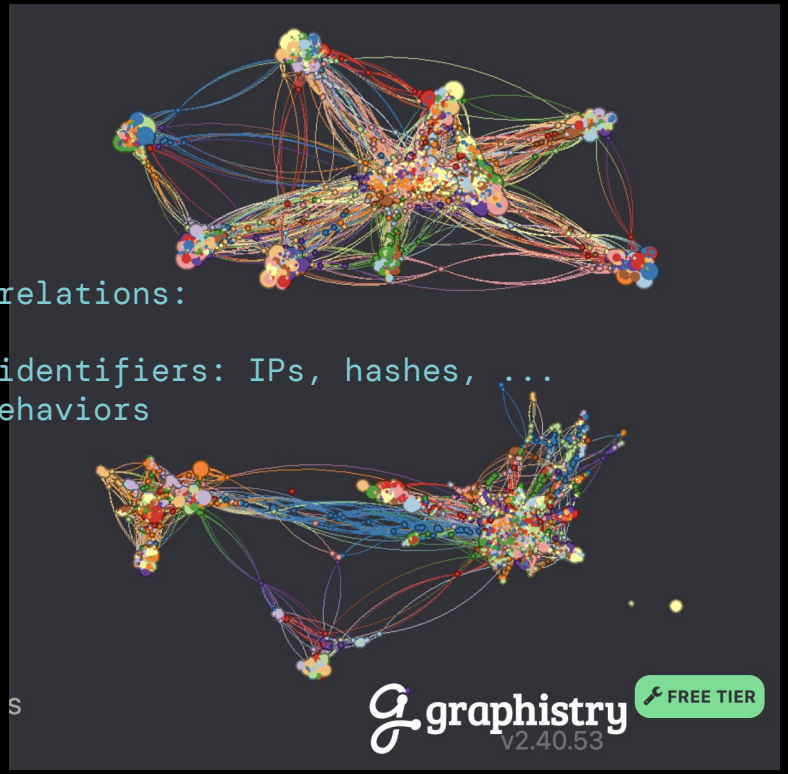
	DetectTime	Target	ConnCount	Format	Category	Source	WinStartTime	WinEndTime	ID	ByteCount	FlowCount	PacketCount	CreateTime	EventT
60822	2019-03-14 10:21:29+00:00	NaN	NaN	IDEA0	Malware	[{"Hostname": ["hostname367"]}]	NaT	NaT	6a03bb8c-023a-461b-9752-29b330df9d1c	NaN	NaN	NaN	NaT	2019-0 10:21:00+0
125167	2019-03-16 18:21:39+00:00	NaN	NaN	IDEA0	Malware	[{"IP4": ["71.33.49.149"]}]	NaT	NaT	ec6a43bd-6bdb-4c5b-a6b5-790ffc162542	NaN	NaN	NaN	NaT	2019-0 18:21:00+0

Data are Correlations; use Graph AI to see, understand and predict

time	src_domain	src_computer	dst_computer
150885	U620@DOM1	C17693	C1003
151036	U748@DOM1	C17693	C305
151648	U748@DOM1	C17693	C728
151993	U6115@DOM1	C17693	C1173
153792	U636@DOM1	C17693	C294
...	...	...	...
48263	C11843$@DOM1	C11843	C528
77937	C8470$@DOM1	C8470	C528
173300	C716$@DOM1	C716	C716
102472	U7365@DOM1	C16126	C586

Two kinds of correlations:

- Strongly link identifiers: IPs, hashes, ...
- Fuzzily link behaviors



Seeing the Attack Surface

Demo:

1. Natural language querying of Splunk & multi-step actions

2. Automatic UMAP embedding + visualization of rich table

https://www.loom.com/share/0b598a3ddec94b9b8ee9f5eb94466ab4?sid=b5aba3dc-5d80-4f0e-8538-c863565522e9

More information:
See Graphistry talk at NDC Security 2024

Use Graphistry directly in Databricks

- Augment SIEM with security data lake + AI
- Combine with Splunk/Snowflake
- Unify views and reduce noise via queries
- Graphistry[ai] directly in databricks

Graphistry[AI]

- GPU Graph intelligence: Viz, querying, & AI <-- databricks is mostly tabular
- Turning Logs into Signals: AVR
- Analysts: Hunting UI across DBs <-- analyst can combine data in splunk + sql in same workflow
- Managed investigations: Continuously learning security workflows
- End to End GPU pipeline using CuCat (Spark Databricks too)

Louie.AI progress on speed running the challenge

1. PyGraphistry[AI]: Automates & GPU accelerates

2. Louie interfaces: Integrates notebooks, dashboards, automation

3. Louie genAI: Conversational interfaces, continual learning,
   + LLM text analysis

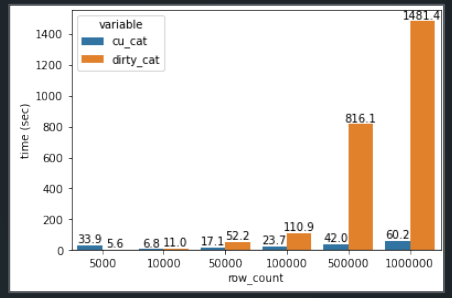


Fig: GPU cu_cat for feature engineering

Close to achieving the full speed run

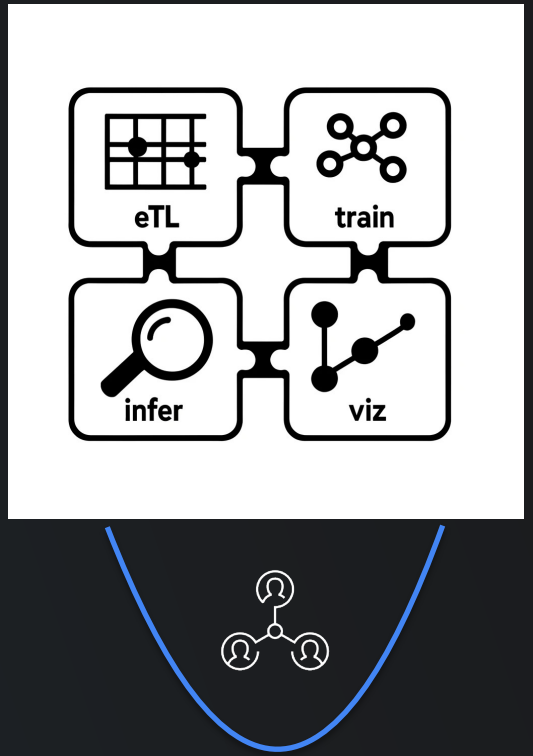
Road Map

- Graphistry 2022
- pyGraphistry[AI]
- Louie 2023
- Data schema learning
- Tools, connectors, process models
- Python sandbox
- pyGraphistry[AI] agent: graph, umap, …
- cu_cat feature engineering
- GFQL graph dataframe query language
- Louie 2024 Q1
- Dashboarding
- GPU runtime
- Pattern learning
- Louie 2024 Q2
- Automation

Ultimately this is a talk about the EDA process — mission critical in an hour

Modern EDA

What we want



Modern EDA with Graphistry


```
g = graphistry.nodes(df)
```


```
g.umap().dbscan()
```

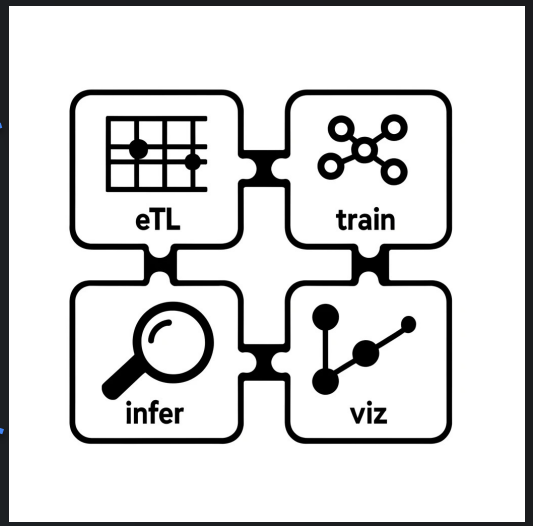

```
g.transform(df_batch)
```


```
g.plot()
```

… lots more, ask us about auto GNNs, GFQL, self writing wikis and KGs

Full AI & Viz Suite

Permissioned
Shared
Embeddable



Acceleration for Scale
(CuCat)
(eg, terabytes come talk to us)

How does this work? Feature Extraction and AutoML

time	src_domain	src_computer	dst_computer
150885	U620@DOM1	C17693	C1003
151036	U748@DOM1	C17693	C305
151648	U748@DOM1	C17693	C728
151993	U6115@DOM1	C17693	C1173
153792	U636@DOM1	C17693	C294
...	...	...	...
48263	C11843$@DOM1	C11843	C528
77937	C8470$@DOM1	C8470	C528
173300	C716$@DOM1	C716	C716
102472	U7365@DOM1	C16126	C586

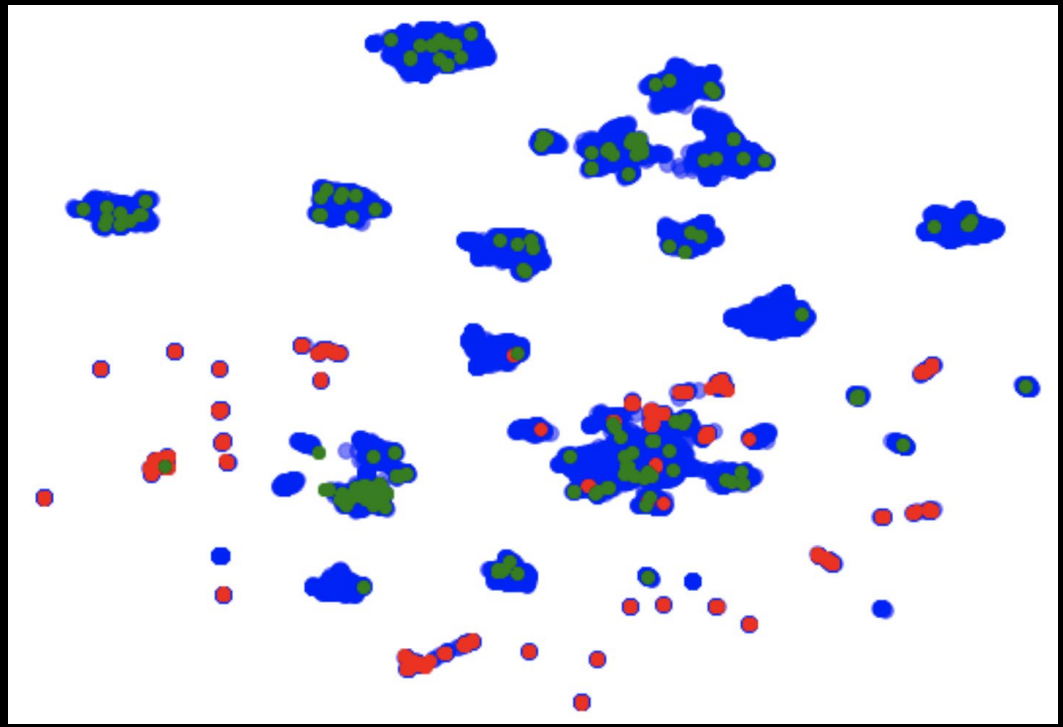
```
# see how the model has organized features
X = g5._node_features
X
```

	feats: c9990, c9994, c9997	feats: kerberos, u1, u7	feats: c528, c5252, c5281	feats: c612, c6121, c6125	feats: c2446, c2444, c24464	feats: c13713, c13130, c13134	feats: c586, c5866, c5864	feats: c467, c4674, c4667	feats: unlock, c1111, c11114	feats: c625, c6257, c6255	...	feats: c5299, c529, c5294	feats: c16616, c16168, c16663
0	0.052992	0.050029	0.051413	0.051403	0.050019	0.061212	0.051419	0.050463	0.057304	0.051460	...	0.051392	0.063923
1	1.609871	0.050030	0.051420	0.051410	0.050016	0.091486	0.051426	0.050465	0.061034	0.051467	...	0.051399	0.069057
2	0.051975	0.050030	0.557747	0.054309	0.050016	0.070115	0.051457	0.050475	0.060911	0.051500	...	0.051429	0.068827
3	0.052089	0.050032	0.051534	0.051523	0.050023	0.069080	0.051541	0.050501	3.781985	0.051586	...	0.051511	0.074502
4	1.612539	0.050031	0.051482	0.051472	0.050016	0.070362	0.051488	0.050485	0.061027	0.051532	...	0.566014	0.069057
...	...	...	...	...	...	...	...	...	...	...	...	...	...
19008	0.051856	22.477729	7.183005	0.051355	0.050015	0.061363	0.065023	0.050447	2.851467	0.051411	...	0.080029	1.255318
19009	0.051961	0.077069	5.711064	0.051431	0.050016	0.051497	0.069693	0.050472	0.050832	0.051490	...	0.093134	0.051501

Using pygraphistry[AI] to extract features (from logs, anything dataframe-y)

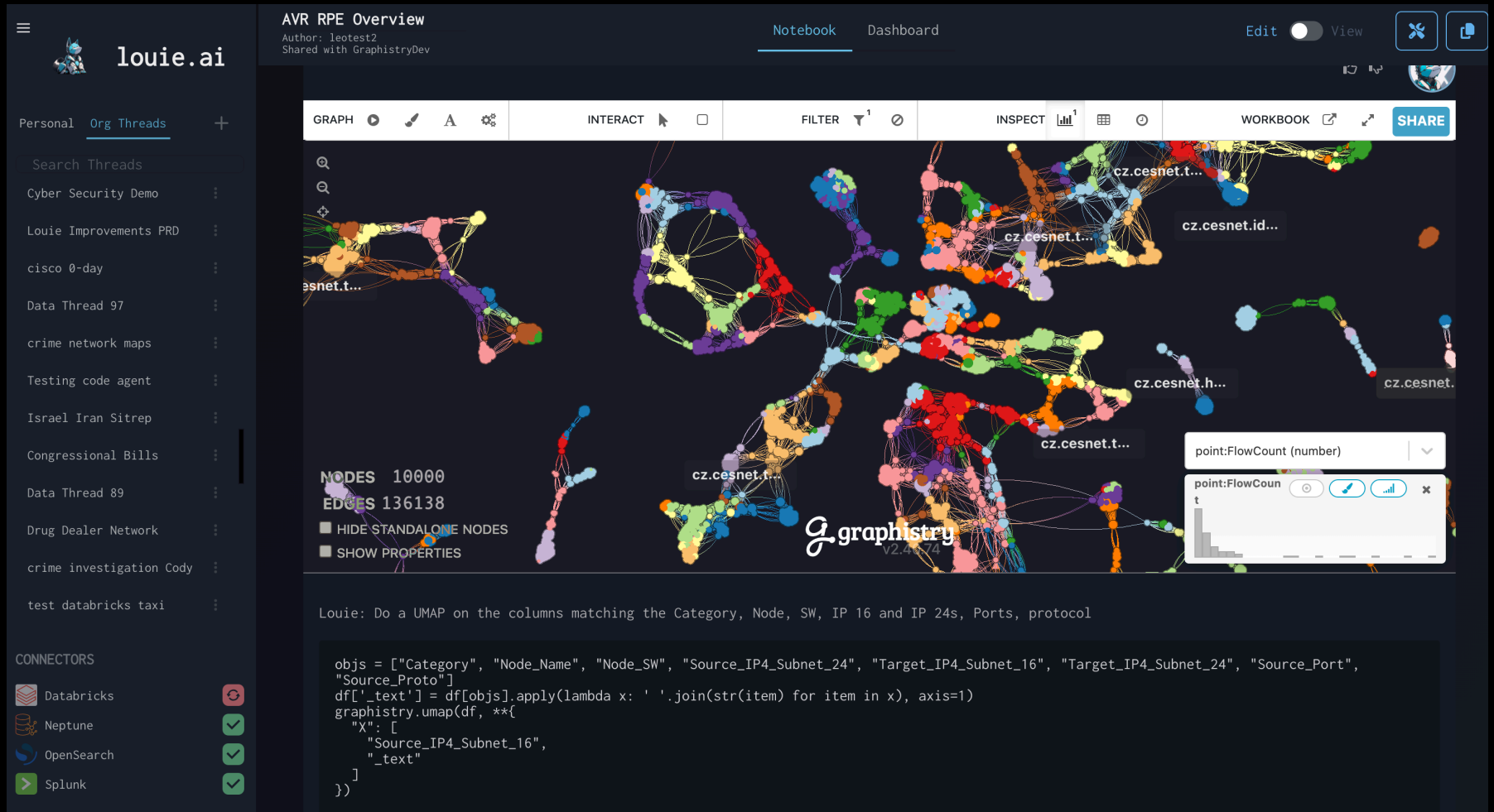
Predict behavioral correlation id given the intrusion pattern -> incident id

time	src_domain	src_computer	dst_computer
150885	U620@DOM1	C17693	C1003
151036	U748@DOM1	C17693	C305
151648	U748@DOM1	C17693	C728
151993	U6115@DOM1	C17693	C1173
153792	U636@DOM1	C17693	C294
...	...	...	...
48263	C11843$@DOM1	C11843	C528
77937	C8470$@DOM1	C8470	C528
173300	C716$@DOM1	C716	C716
102472	U7365@DOM1	C16126	C586



Seeing the Attack Surface - Predictive Modeling

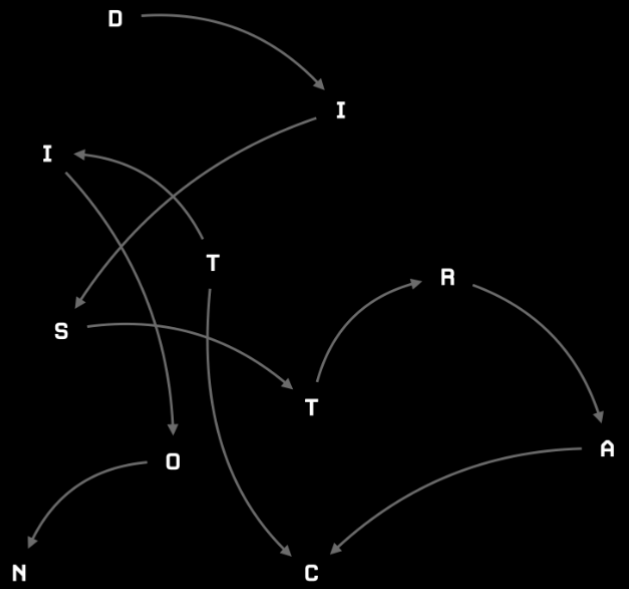
Can it get Easier?



Louie Optimizes Path Traversal

The Modern SOC Analyst Struggle

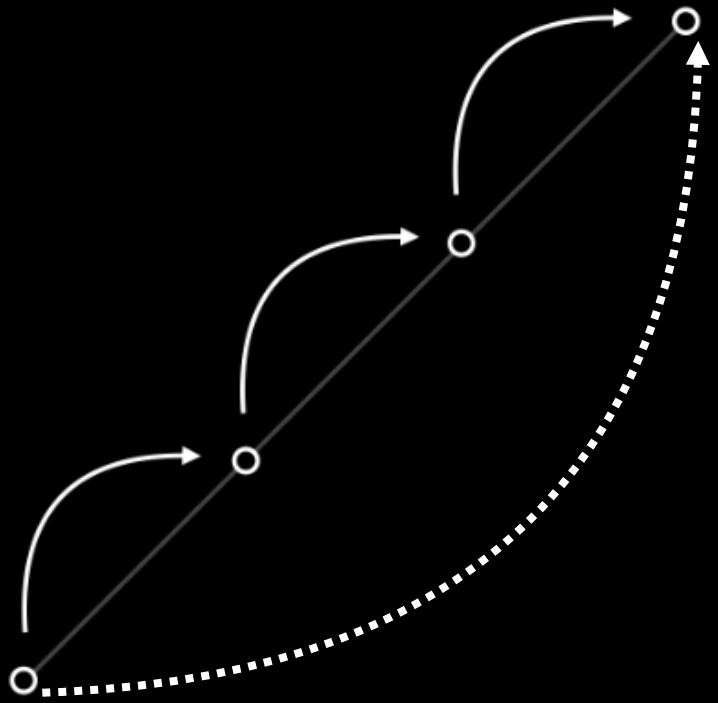
- Investigations take a Tree structure
- Louie learns optimal path for reusable playbooks/plans
- Agentic-OS you connect with your own flows
- Learning Loops so that python/sql/tickets aren't one offs, but feed back into system



Louie helps you turn Process into Plans

SOC's are full of Context and Expertise

- Investigations are steps
- Highlights Process vs Planning
- Optimizes and personalized user experience
- Build reusable pipelines, automatically



Process vs Plan

Louie.AI: GenAI learning operating system for teams, data, & code

AI analyst notebooks

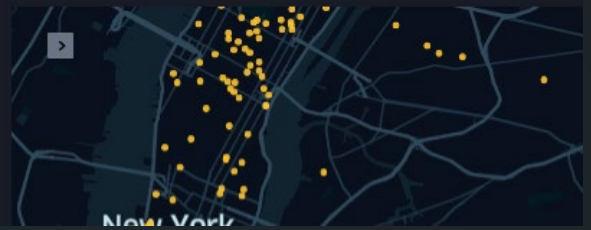
```
get 100 taxi rides

| search index=nyc_taxi | head 100 | fields *
```

VendorID	dropoff_latitude	dropoff_longitude	fare_amount	host
1	40.76208496	-73.97861481	6	splunk.graphistry.com
1	40.75984192	-73.98239136	5.5	splunk.graphistry.com

AI dashboards | AI pipelines | AI APIs

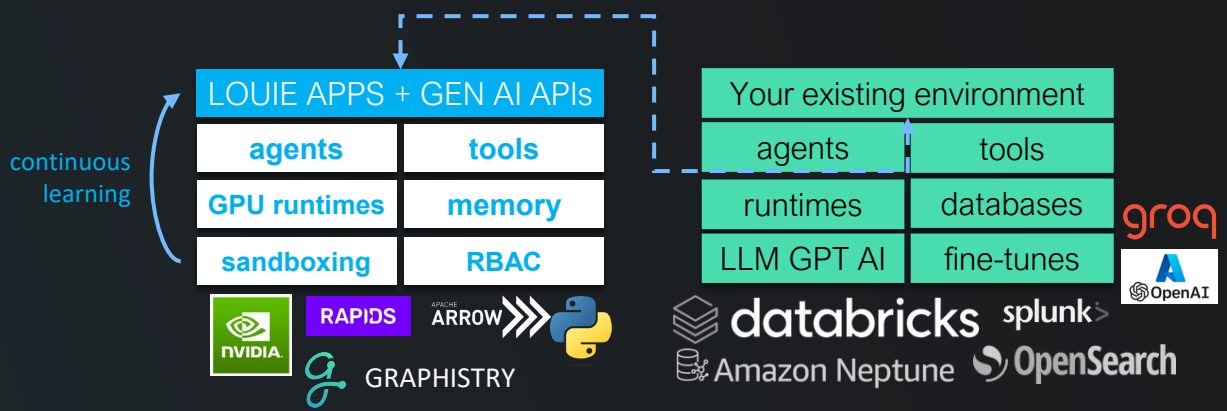
Dataframe @2cf1035a - 100 rows, 28 columns

Make a geo map visualization



Louie learns & talks to:

Databases & APIs: SQL, Log, Graph, …
Indexes data: Wikis, …
Data science & coding libraries
How your team works
Plug in your own LLM, Agent, Data, API, …



Builds on Graphistry's GPU visual graph AI platform

Ask about connectors, APIs, roadmap, & innovation projects

Thank you!

Takeaways

- GenAI reset happening, with authoring stack first
- Look for learning loops + data pipelines
- Supercharges mission critical investigations in real time (CYBERCOM, supply chains, knowledge graphs, etc)
- Scaling & autonomy happening, longer timeline

```
! pip install pygraphistry
```

- GPU graph viz
- UMAP
- cu_cat
- GFQL

Contact @ louie.ai for Early Access Program

Dr Alex Morrise
@silkspaceships