

Twitter Thread by Philip Vollet

Philip Vollet

@philipvollet



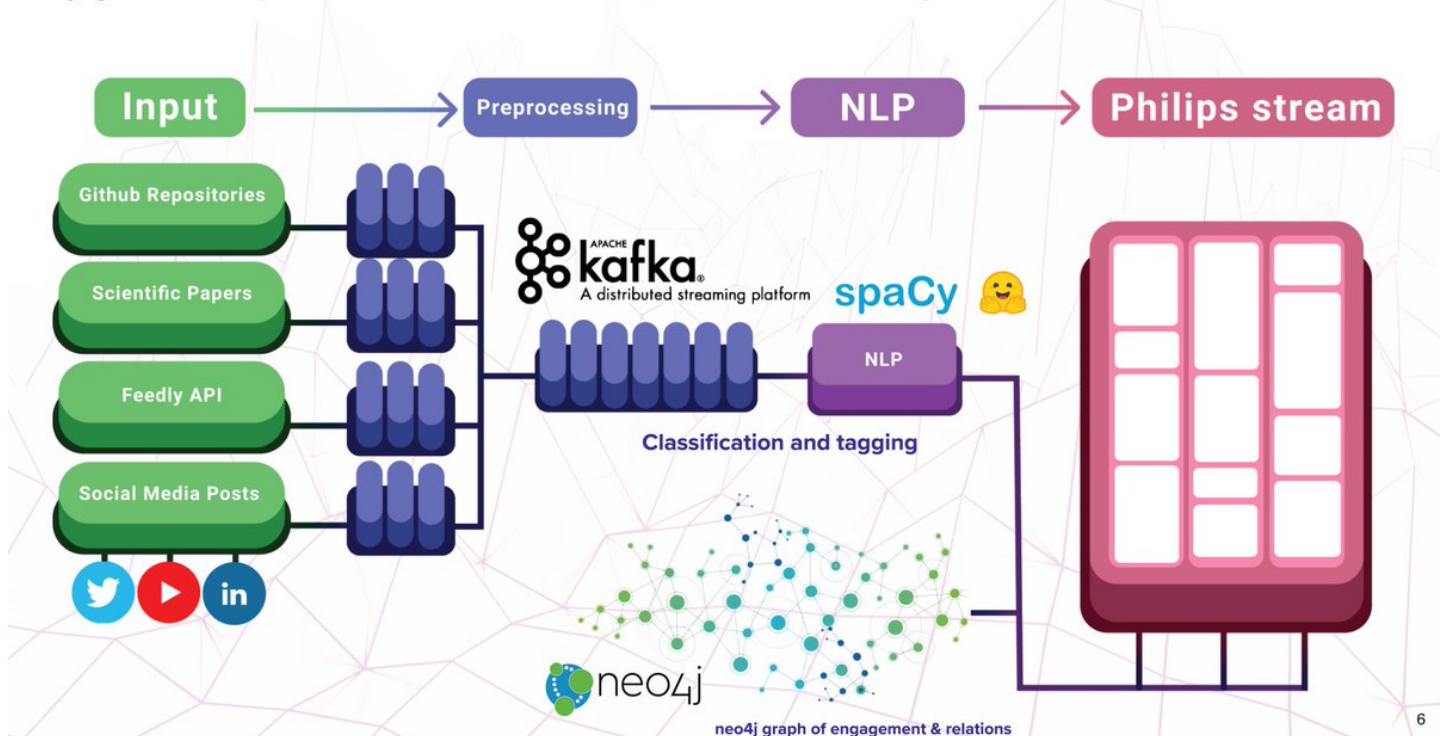
Insights from an open source influencer

I'm often asked how I get my content, over the years I've built an unusual technology stack for it

<https://t.co/FdIAeZc8GF>

Some insights:

My goal is unique content and to be fast, for this I have a special infrastructure



I use Feedly for most content inputs because I can access the content through a single API endpoint and scraping is often pure pain.

[@feedly](https://twitter.com/feedly)

Feedly saves me a lot of time and manual work.

<https://t.co/CeHju43rwJ>

To pull and enrich my GitHub content I use ghapi from [@fastdotai](#) which provides a 100% always-updated coverage of the entire GitHub API

<https://t.co/c0HwjoShvg>

The actual magic happens on my server infrastructure.

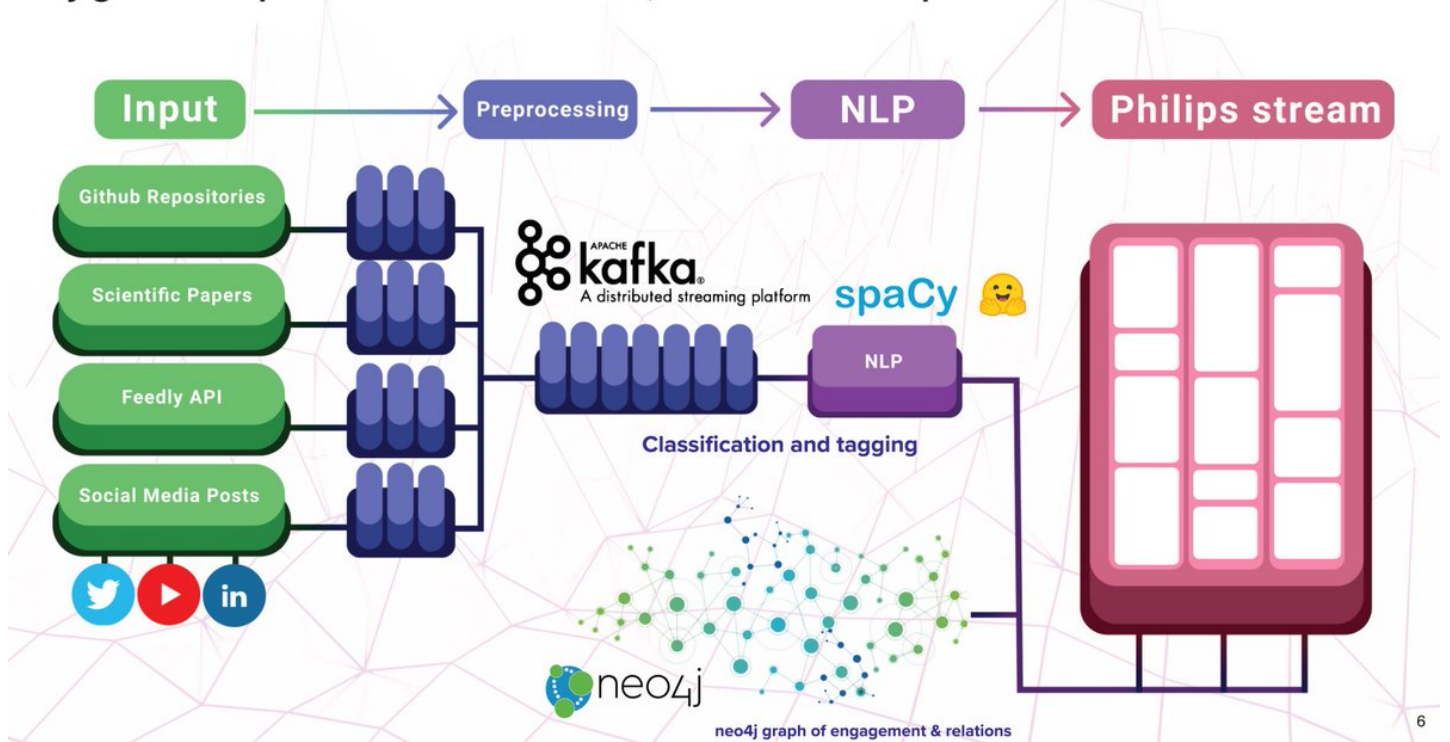
This is where the various data streams converge and are enriched.

The whole thing is based on Kafka streams so that if a pipeline stops it can be started up again without any problems.

The image displays a collage of terminal windows from a server environment. The top row shows two instances of the 'pv' project, each with a 'Containers' section listing running containers like 'kafkadrop_kafdrop_1', 'kafkadrop_kafka_1', 'kafkadrop_kafka-server2_1', 'sleepy_gould', 'kafka-server1', 'tender_northcutt', 'zookeeper-server', and 'neohype'. Below the containers, there are graphs for CPU usage (0.76 and 0.56) and Memory usage (1.21). The middle row shows system status for 'neohype' (Ubuntu 18.04) with CPU at 22.3%, MEM at 20.7%, and SWAP at 0.1%. It also displays a 'CONTAINERS' table with columns for Name, Status, CPU%, MEM, /MAX, IOR/s, IOW/s, Rxs/s, Tx/s, and Command. The bottom row shows Kafka logs with messages like 'INFO Replica loaded for partition linkedin-insights-0 with initial high watermark @ (kafka.cluster.Replica)' and 'INFO [Partition linkedin-insights-0 broker=0] linkedin-insights-0 starts at Leader Epoch 0 from offset 0. Previous Leader Epoch was: -1 (kafka.cluster.Partition)'. There is also a Redis terminal window showing 'Redis 6.0.3 (00000000/0) 64 bit' and 'Running in standalone mode'.

Then a wild mix of machine learning NLP pipelines between [@huggingface](#) and [@spacy_io](#) is used to classify, score and tag the content.

My goal is unique content and to be fast, for this I have a special infrastructure

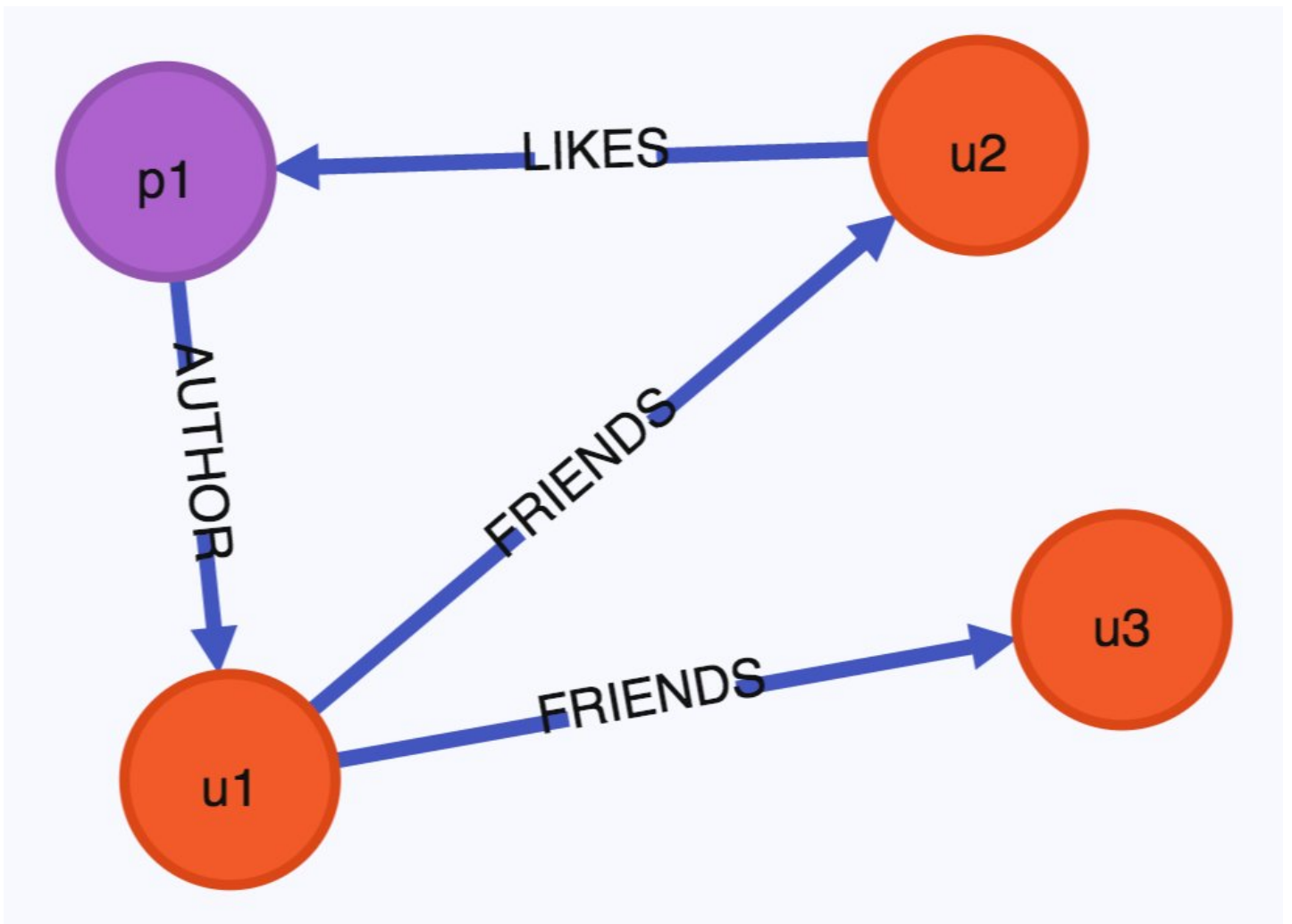


For analysis I have a [@neo4j](#) instance to analyze my network relations and the engagement.

This is also used to find potential new sources.

Why?

Because it's about engagement, influence and adding value



Spread the open source love!

If you know an amazing project drop me a message
[@philipvollet](https://twitter.com/philipvollet)

<https://t.co/FdlAeZc8GF>