



Figure 1: Git-approach to bring light into our scrollable, short-lived world

Abstract

Let's enumerate in one section what this is all about (yes, we do repeat Personal Knowledge Library (PKL) definition on purpose):

- **Overarching Goal:** A social-network consisting of Knowledge Graphs in the form of Personal Knowledge Libraries (PKL)
- **Public Section of Personal Knowledge Library (PKL)** as a way to build up knowledge adhering to "standing on the shoulders of giants"
- **Git-Approach to Personal Knowledge Library (PKL)** adheres to "cross-validation" principles, by forking out, reassessing and making a pull-request/ merge back in the original Personal Knowledge Library (PKL)
- By visualizing the intellectual trajectories of thought and discovery in the Public Section of the Personal Knowledge Library (PKL), we enable some kind of "reproducibility"
- Use The Wire-Box [\[link\]](#) or Augmented Argumentation via Agent Interactions to encapsulate expert knowledge and an infinite universe of further options
- Plot and re-use a flawed reward system

The Problem

Today's "infinite scroll"/"slot-machine" presents content—whether posts, images, or videos—in a seemingly endless, disconnected stream. This disconnect impedes users from delving into interconnected topics, depriving them of a cohesive understanding. Furthermore, most of these platforms and even more scientific ones e.g., nature journal, are triggered by the rewards of quick wins & news coverage. Such superficial and short-lived scrolling news contrasts sharply with Knowledge Graphs, where interconnected pathways usher users to deeper insights and to track the history & performance of news-commits.

Solution Pillars

- **Personal Knowledge Libraries (PKL):** Example: Imagine Instagram's chronological photo feed, but instead of photos, you have nodes representing key thoughts, ideas, or memories. Or consider LinkedIn, where connections don't just represent professional relationships, but conceptual links between ideas. In essence, PKL isn't just another social media platform—it's a fusion that amalgamates the essence of Instagram's visual storytelling, Twitter's/ X's bite-sized thoughts, LinkedIn's professional sharing, and even Tinder's interactive engagement, all aimed at charting the growth of a person's intellectual journey.

Immediate Goals

One immediate goal is to design a platform for a knowledge graph generation with a simple, maintainable pipeline for content (blobs) to graph integration. This includes modules such as 'Time-Stamped Content Evolution Graphs' and 'Augmented Argumentation via Agent Interactions,' with the option to segregate between private and public repositories, allowing users to control what they share.

Future Vision

Pair PKL with virtual reality to visualize a personal 3D knowledge universe, creating an immersive narrative of evolving thoughts. Imagine navigating a cerebral museum, where every node and connection brings to life the intellectual universe. This could aid others facing similar problems and life stages.

Time-Stamped Content Evolution Graphs

Consider how narratives such as 'A Dangerous Game' and 'Arrival' challenge conventional understandings of time. These works suggest a more fluid conception of time, which invites a re-evaluation of our experiences. Politicians' shifting speeches and the prevalence of 'Fake News' complicate discerning fact from fiction. Our git-versioned content graph system records 'commits' of content from specific sources at specified times, allowing users to trace narrative evolution and recognize patterns.

Current Status: MVP in Development.

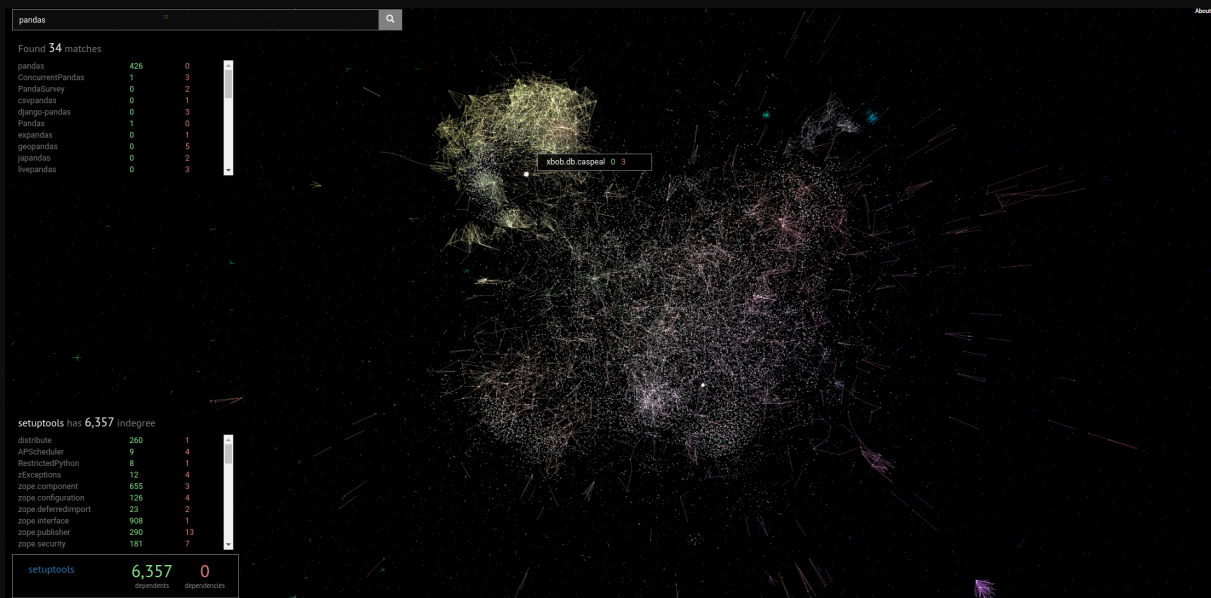


Figure 2: View on "Code Galaxies Visualization" by Andrei Kashcha

The Wire-Box or Augmented Argumentation via Agent Interactions

Envision a box of wires, akin to neural connections in our brain. These represent the edges, links, and pathways of thoughts that users can customize. Users can segment their thought processes, encapsulate them within agents, and interlink them. For example, tracking a politician's statements on climate change could involve agents for environmental data, socio-economic implications, and global political reactions. These agents are interconnected facets of understanding, mimicking internal dialogues for complex topics. Current Status: MVP Available [\[here\]](#)(link to MVP).

Why the Wire-Box

The Wire-Box was designed with the vision of enabling non-technical users to utilize graph-based agent behavior. It's not just about using expert knowledge but allowing experts in various fields to apply their expertise in a meaningful way. The idea is to enable experts to unlock knowledge encapsulated in language corpora used by LLMs through customized instruction and layering of agents, using graphs in conjunction with agents for robust, reliable, and reproducible outcomes.

Business-Model for Sub-Module Wire-Box

Business-Model:

```
sequenceDiagram
    participant U as User
    participant S as System
    participant DB as Database
    participant C as Company
    participant E as Experts
    U ->> S: Create agent/agent pool
    S ->> DB: Save agent/pool
    U ->> S: Deploy/Redeploy/Layer agent
    S ->> DB: Update agent/pool status
    loop Validation Process
        S ->> S: Validate and optimize agent connections
    end
    S ->> C: Rent out robust agents
    C ->> U: Provide maintenance
    U ->> E: Experiment with agents (sandbox)
    E ->> S: Feedback and improvements
```

Description:

- **Agent/Agent Pool Creation:** Users (U) initiate the process by creating agents or agent pools based on their (expert) knowledge.
- **Saving to Database:** Once an agent or agent pool is created, the System (S) saves this information in a Database (DB). This step ensures that all agent data and configurations are securely stored and managed.
- **Deployment and Layering:** Users have the ability to deploy, redeploy, or layer their agents. This means they can initiate the agent, make modifications, or combine it with other agents or pools to enhance its capabilities or efficiency.
- **Status Update and Validation:** As agents are deployed or modified, the System updates their status in the Database. Concurrently, there's an ongoing validation process where the System continually validates and optimizes the connections and interactions between agents.
- **Renting Out Robust Agents:** Once agents are deemed robust and efficient through the validation process, the System (representing the Company, C) rents them out. This implies that other entities or customers can use these well-developed agents, likely for a fee.
- **Maintenance and Experimentation:** While the Company is responsible for the maintenance of these agents, ensuring they run smoothly and effectively, Users are given the opportunity to experiment with these agents in a sandbox environment. This environment allows Users to test and experiment without affecting the live, operational versions of the agents.
- **Feedback Loop with Experts:** Experts (E), who might be more advanced users or specialists in the field, use these agents and provide feedback and suggestions for improvements to the System. This feedback is crucial for the ongoing development and refinement of the agents.