

VisPile: A Visual Analytics System for Analyzing Multiple Text Documents With Large Language Models and Knowledge Graphs

Adam Coscia
 Georgia Institute of Technology
acoscia6@gatech.edu

Alex Endert
 Georgia Institute of Technology
ender@gatech.edu

Abstract

Intelligence analysts perform sensemaking over collections of documents using various visual and analytic techniques to gain insights from large amounts of text. As data scales grow, our work explores how to leverage two AI technologies, large language models (LLMs) and knowledge graphs (KGs), in a visual text analysis tool, enhancing sensemaking and helping analysts keep pace. Collaborating with intelligence community experts, we developed a visual analytics system called VisPile. VisPile integrates an LLM and a KG into various UI functions that assist analysts in grouping documents into piles, performing sensemaking tasks like summarization and relationship mapping on piles, and validating LLM- and KG-generated evidence. Our paper describes the tool, as well as feedback received from six professional intelligence analysts that used VisPile to analyze a text document corpus.

Keywords: Visual analytics, Sensemaking, Text analytics, Large language models, Knowledge graphs.

1. Introduction

Performing visual text analysis is an ongoing challenge in the visual analytics community, where reading, understanding, and making sense of large amounts of text is time-consuming. As data scales increase, decision-makers in the intelligence community seek new methods to balance the constant tension between performing human and automated analysis (ODNI, 2019; Sayler, 2020). Automated methods, including AI, can help extract and summarize relevant text and reduce the need for extensive reading, while human-in-the-loop interfaces help people combine visual and analytical methods to contextualize evidence and synthesize critical insights (Baker et al., 2009).

In visual analytics, two recent technologies have dominated the discourse: large language models (LLMs) and knowledge graphs (KGs). LLMs can rapidly summarize text, explain concepts, and answer

questions (Srivastava et al., 2023). KGs can validate ground-truth information generated by LLMs and link entities and events back to source text (Li et al., 2024a). Recent visual analytics tools are exploring how LLMs and KGs can assist in task planning, question-answering, and summarization of unstructured text. LLMs and KGs could help analysts distill large document sets into relevant data and synthesize new evidence, while minimizing the need for detailed manual reading. However, understanding how to effectively integrate these technologies into visual text analytic tools is less well understood.

In this work, we explore how LLMs and KGs can be integrated into visual text analysis through a yearlong design study with domain experts in the intelligence community. While LLMs and KGs show promise in automating time-consuming analysis tasks, it is unclear how to design a visual analytics tool around their emergent capabilities. To address these gaps, we present **VisPile** (Sect. 3), a visual analytics system for analyzing text documents. Our system incorporates LLM and KG features to help analysts group documents into piles and perform sensemaking tasks (summarization, extraction, Q&A, etc.) on piles. We demonstrate how LLM and KG features in VisPile impact sensemaking through formative feedback from six professional intelligence analysts (Sect. 4). The analysts used the LLM and KG to quickly find and compare relevant subsets of 845 documents for piling, chained LLM tasks and KG facts together to enable deeper analysis of their piles, and combined the LLM response and KG suggestions to contextualize evidence and uncover hidden connections.

In summary, we contribute: (1) design goals for integrating LLMs and KGs into visual text analysis; (2) **VisPile**, an open-source¹ visual analytics tool with LLM and KG features for text document analysis; (3) domain expert feedback illustrating preliminary results on how LLMs and KGs impact sensemaking in text analysis.

¹VisPile code: <https://github.com/AdamCoscia/VisPile>

2. Background and Related Work

We first describe the intelligence analysis process and related visual analytics systems, then recent usage of LLMs and KGs in visual text analysis.

2.1. Visual Analytics for Intelligence Analysis

Our designs are grounded in the Pirolli and Card sensemaking loop for intelligence analysis (Pirolli and Card, 2005). In a bottom-up sensemaking process, intelligence analysts iteratively schematize information spread across a corpus of text documents like news articles. To gain situational awareness, analysts usually group related documents into piles, grounding their understanding of both relevant and irrelevant information (Shipman and Marshall, 1999). With piles of documents at hand, analysts read through them to gather reportable evidence, usually snippets describing key people or events related to the pile’s topic. Finally, raw evidence needs to be verified, typically by mapping it to documents in and across piles.

Research in visual analytics has a rich history of developing techniques for sensemaking over unstructured text data like documents (Endert et al., 2014; Kang and Stasko, 2011; McColgin et al., 2006). For example, interactive topic modeling (Berry and Kogan, 2010) lets users guide ML models to identify documents relevant to bottom-up sensemaking. Tools like TopicSifter (Kim et al., 2019) enable human-in-the-loop feedback by allowing users to target documents to guide the search process. Spatial analysis methods have also proven effective for text document sensemaking (Andrews et al., 2010; Endert et al., 2012). Tools like Analyst’s Notebook (i2 Group, 2024) support evidence marshalling by organizing documents into related groups. Other systems span multiple stages of sensemaking; e.g., Jigsaw (Stasko et al., 2007) uses multiple coordinated views to reveal relationships across documents. Recent works have begun leveraging LLMs and KGs in visual analytics. KNNowNEt (Yan et al., 2025) uses a KG to enhance LLM-driven question-answering but does not support multi-document analysis or evidence marshalling. LEVA (Zhao et al., 2025) augments visual analytics systems with automatic insights and report generation but lacks controls for selecting sensemaking tasks.

2.2. LLMs & KGs For Visual Text Analysis

LLMs and KGs have individually demonstrated impressive data analysis capabilities, inspiring us to consider their potential to support the intelligence analysis process. Specifically, we focus on the potential

for LLMs and KGs to help analysts retrieve relevant text documents and perform sensemaking over them.

LLMs demonstrate comparable performance to humans in fetching, parsing, and visualizing tabular data autonomously (Cheng et al., 2023; Zhang et al., 2024). However, the space of text document analysis is less explored. Classification models like BERTopic (Grootendorst, 2022) can generate open-ended topic models from a corpus of text documents, providing an entry point into exploring the document space when little is known up-front. Alternatively, generative models like OpenAI’s GPT-4o can perform a semantic similarity search using retrieval-augmented generation (RAG) techniques (Lewis et al., 2020) to return relevant documents based on an open-ended query. Given a group of documents, LLMs can perform tasks like entity extraction, summarization, and question-answering using prompts to guide the LLM’s output (Liu et al., 2023). LLMs could help analysts shift time spent away from reading and towards synthesizing information.

Conversely, KGs are a more nascent technology in the text analysis space. KGs encode semantic relationships between entities as triples in the form subject→predicate→object, with metadata on the nodes (subject, object) and edges (predicate) as properties (Li et al., 2024a). Consider the triple John→likes→Sally, where John and Sally might be labeled as Person type and likes as Emotion type in their metadata. Their semantic nature makes them particularly suited for providing context to text content, e.g., text generated by an LLM (Hogan et al., 2021). Most prior KG systems focus on direct question-answering (Li et al., 2024b) or visual exploration of existing graph structures like Wikipedia (Latif et al., 2021). However, recent work has explored using LLMs to extract triples and create a KG directly from a document corpus (Pan et al., 2024). We utilize this technique in our work to open the door for exploring new text analysis methods using the KG.

3. The VisPile System

We present **VisPile**, a visual analytics tool for text document analysis. VisPile integrates an LLM and a KG into searching, filtering, and piling documents, analyzing documents in piles, and validating LLM- and KG-generated evidence. Users combine LLM and KG features in VisPile into various sensemaking workflows.

Throughout our system description and expert evaluation, we used the IEEE 2014 VAST Challenge dataset (KRONOS) as a proof-of-concept (Whiting et al., 2014). The dataset includes 845 plain-text news reports (500–1000 words each) describing complex relationships that culminated in a kidnapping on the

fictitious island nation of Kronos. The central task is: “*Synthesize potential relationships that may have led to the disappearance.*” The KRONOS dataset is widely used as an unclassified proxy for intelligence analysis. We also used OpenAI’s GPT-3.5 Turbo, the state-of-the-art generative LLM at the time, to process data and run prompts. Our data architecture supports any similarly structured dataset and generative LLM, including open-source models. All prompts are available in the repository and supplemental material.

3.1. Design Goals

To develop VisPile, we followed the methodology for visualization design studies proposed by Sedlmair et al. (2012). Our team comprised the authors, visualization experts in human-centered computing and data analysis, working with professional SIGINT analysts, program coordinators, and managers with years of combined experience working in the U.S. Intelligence Community (IC). We first identified where LLMs and KGs could enhance stages of sensemaking in intelligence analysis. We then refined what LLMs and KGs would do and how analysts would interact with them through bi-weekly focus groups. In each session, the visualization experts presented low-fidelity mockups to gather and refine iterative design feedback. Co-designing with domain experts in both visualization and the IC helped us synthesize shared design goals towards enhancing the sensemaking process.

We established a shared technical terminology (Fig. 1, top) based on the analysts’ workflows, used throughout the paper. To scope our investigation, we focus on analyzing text-only **documents** like technical reports or news articles. Analysts often organize their reading by manually arranging documents into **piles**, groups of documents that share a common topic or purpose. As analysts read documents, they often perform entity-based analysis, searching for **facts** that represent the relationship between two entities; e.g., “John likes Sally.” They may also generate intermediate artifacts like summaries or answers to questions as they work, which we refer to collectively as **evidence**.

In VisPile, LLMs and KGs are the primary method of finding and grouping relevant documents into meaningful piles, using piles to extract facts and evidence from documents, and analyzing facts and evidence for insights. This represents a fundamental shift in the intelligence analysis process, where fact/evidence gathering is traditionally a cognitive process of the analyst only. The LLM fulfills the role of synthesizing massive amounts of text quickly to generate/answer questions and summarize relevant

entities, relationships, topics, facts, etc. KGs, on the other hand, already represent the distillation of text documents into a set of interconnected facts. Analysts can traverse facts in the KG directly, only reading documents when necessary, as well as compare LLM-generated evidence with known KG facts.

Piles are vital for scoping the data that automated functions are allowed to operate on. VisPile enables direct manipulation of piles in a spatial view, where analysts drag documents directly into different areas of the workspace and arrange the ordering of piles. The underlying data architecture then handles running LLM and KG functions directly on the documents in piles. Choosing how best to pile documents and analyze piles for the task at hand promotes analyst agency during sensemaking. Our design goals (G) each address aspects of supporting different user tasks within this workflow.

G1 - Leverage LLM for open-ended document search. The LLM should help users forage for documents by recommending open-ended topic groups and interpreting open-ended task descriptions like “*Who are the relevant actors?*” for enabling semantic search.

G2 - Link KG data to source documents. Node and edge metadata should be displayed that links the graph data to source documents, allowing users to easily identify groups of related source documents and create piles from related KG facts.

G3 - Run LLM sensemaking tasks within piles. Users should be able to prompt LLMs to perform analytic tasks on multiple documents in piles. Prompts should be customizable, allowing users to modify and combine operations and even generating custom tasks to target more unique and complex operations.

G4 - Traverse connected KG facts related to piles. Users should be able to interact with and explore the KG visually, connecting facts to their pile for refining their evidence and gathering new evidence from source documents related to their piles.

G5 - Enable validation operations on LLM responses. Users should be able to perform validation operations that keep them in the loop of linking statements and sources to verify the veracity of LLM-generated text. Operations include extracting ground-truth KG data from LLM responses, linking LLM responses to related source documents, and suggesting additional documents relevant to LLM-generated evidence.

G6 - Provide related context around KG facts. Connected and related nodes should automatically be surfaced when traversing the KG, enabling users to contextualize KG facts based on pile evidence.

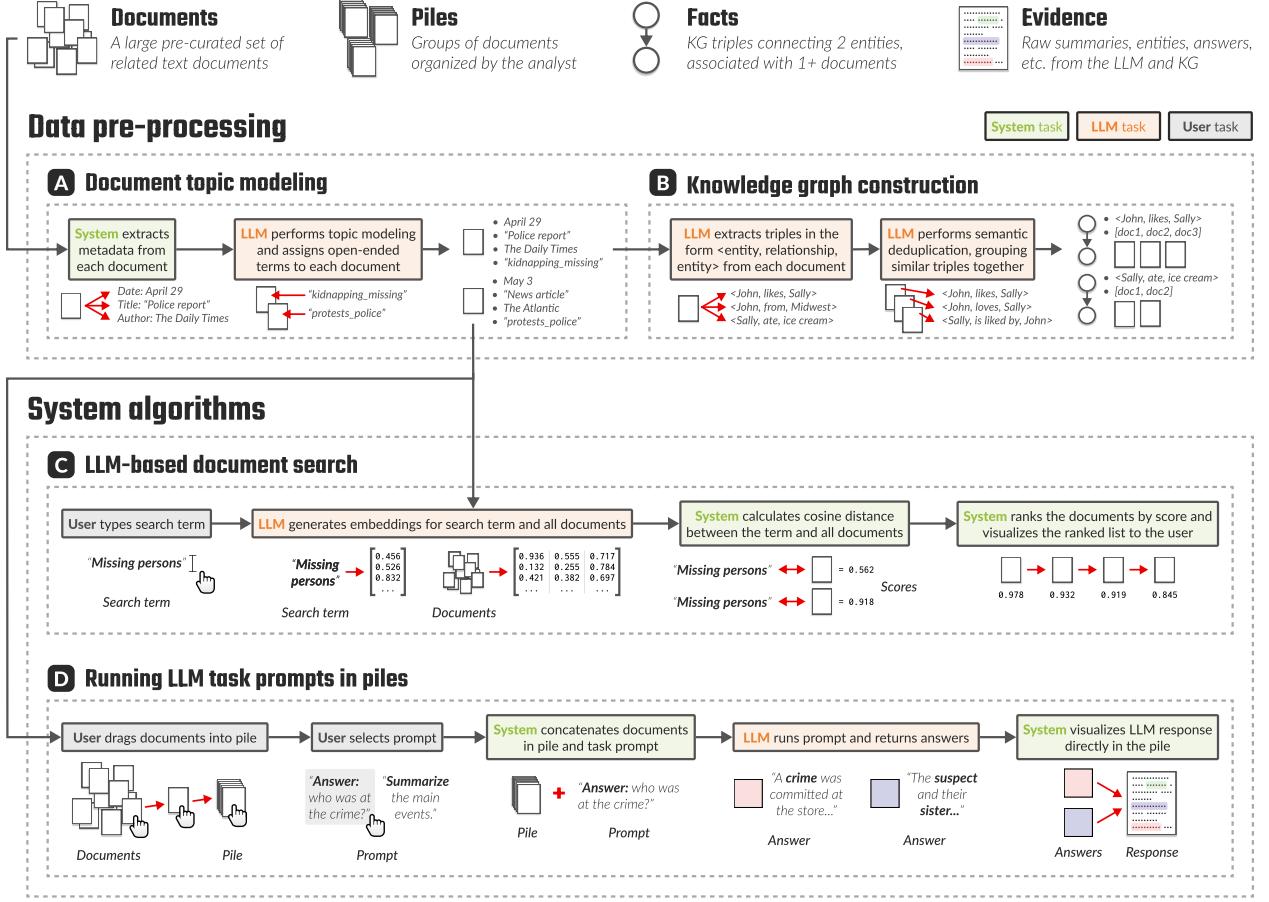


Figure 1. The data architecture for VisPile.

3.2. Data Architecture

VisPile operates within a closed document model, in which a fixed set of related documents is pre-curated but too large to read everything (e.g., ≈ 1000 documents). VisPile takes as input a corpus of plain text documents. Documents are first pre-processed to log metadata such as title, length, and topic. Then, a KG is constructed from the document text, consisting of a list of fact triples with metadata including the source(s) where the fact came from. The list of documents and KG facts are then visualized in a web-based application built with Vue.js.

To generate document topic models (Fig. 1A), VisPile uses BERTopic (Grootendorst, 2022), an LLM-based topic modeling approach. Because LLMs are good at generating open-ended classes, we experimented with not pre-defining topics beforehand and observing how that could improve the piling process. BERTopic produces open-ended semantic topics such as “*kidnapping_missing*” and “*protests_police*”, which are assigned to each document as metadata. LLMs are also used to

perform sensemaking tasks over groups of documents in piles (Fig. 1D). Following Shaib et al., 2023, each task prompt uses text separator tokens to combine documents and the task description into a single prompt. The output from the LLM is then visualized directly in the web app.

To generate the KG, we follow the method proposed by Pan et al. (2024) (Fig. 1B). Similarly to topic modeling, we wanted to explore the benefits that open-ended, LLM-based KG extraction could provide over traditional, schema-based extraction. First, all documents are fed sequentially to a generative LLM (GPT-3.5 Turbo) and raw triples are extracted using prompt engineering (Sahoo et al., 2024) in the form $\langle \text{entity}, \text{relationship}, \text{entity} \rangle$, representing the style of facts commonly used in entity-based analysis in the IC. We then map facts to the subject→predicate→object form of a KG. We save the source document of each triple as metadata. Then, to validate the extracted facts, we run semantic deduplication by grouping repeated facts based on cosine similarity of entity and relationship embeddings. From each group, we select a

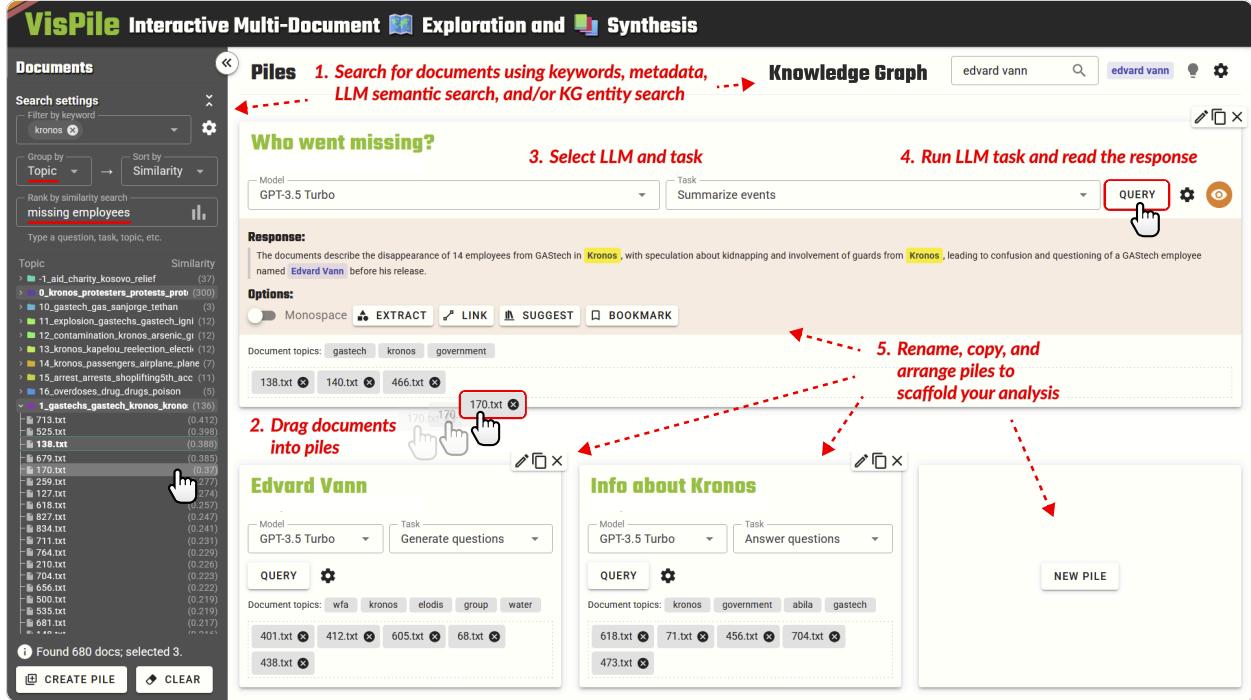


Figure 2. The VisPile interface. VisPile leverages an LLM and a KG to help analysts gather evidence from document collections. The general user workflow is: (1) Search for documents to pile, using keywords, metadata, KG entity search, and/or LLM semantic search; (2) drag documents into piles; (3) choose an LLM and pre-generated task to run as an LLM prompt; (4) Run the LLM task and read the response; (5) Repeat the process, rearranging and renaming piles to scaffold the sensemaking process.

representative fact and record all source documents for the duplicates. This results in a list of non-overlapping triples, each linked to its supporting documents that can be cited. The number of supporting documents serves as a proxy for “support”; i.e., facts cited in more documents may be more trustworthy.

3.3. System Features

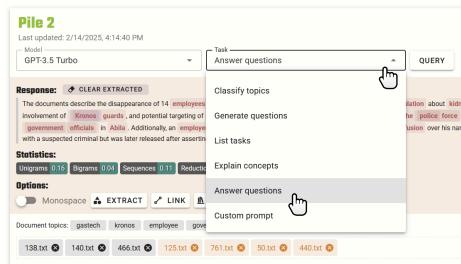
VisPile (Fig. 2) presents several features to support sensemaking: searching and filtering documents using LLM document search and the KG fact list; putting documents into piles and analyzing piles using LLMs and KGs; and validating LLM and KG evidence in piles.

LLM-based document search. Documents in VisPile are shown in a list (Fig. 2, left) that can be filtered using traditional keyword search, as well as grouped and sorted by document metadata such as date, name, and length. VisPile enables open-ended document search (**G1**) in two ways. First, users can organize documents by LLM-generated topic using BERTopic (Fig. 1A), leveraging the strengths of the LLM to take a first-pass over organizing documents by relevant topics. Second, users can perform a

retrieval-augmented generation (RAG)-based semantic similarity search (Fig. 1C). The user’s search query and each document are embedded using a text embedding model (e.g., OpenAI’s text-embedding-3). Document embeddings are then ranked by cosine similarity to the search query. The ranked list is shown back to the user, enabling a more dynamic and iterative search process.

KG fact list. The KG can similarly be used for open-ended document search. Users can access entities in the KG using a free-text search bar (Fig. 2, top). Visualizing large, connected KGs using traditional node-link diagrams can quickly become cognitively demanding (Li et al., 2024a). Instead, VisPile represents the resulting graph around a searched entity as a list of facts (Fig. 3B). The KG fact list shows up to 5 facts, or plain text triples in the form subject→object→predicate. The 5 facts are ranked in the same way as the LLM semantic search (Fig. 1C), by embedding the pile text and all facts, then pairwise comparing them to get a ranked list of top-scoring facts. Source documents for each fact are shown to the right of facts, allowing users to search for documents directly using KG facts (**G2**). Users can click on a document name to filter for that document in the documents view.

A Large Language Model (LLM) tasks



B Knowledge Graph (KG) facts

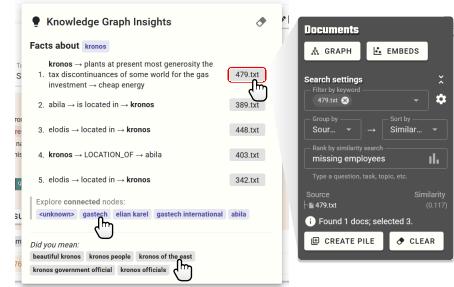


Figure 3. Piles (A) allow users to group documents together and run any of nine sensemaking tasks on the documents as LLM prompts. Prompts are shown for transparency and can be adjusted with inputs for questions, entity types, and concepts, as well as a temperature slider. The KG fact list (B) shows up to 5 top-ranked facts, or plain text triples in the form subject→object→predicate, based on the LLM response in the pile.

Document piles, LLM tasks, and KG traversal. Once a user finds documents they want to analyze, they can drag them into a pile (Fig. 3A) and rename the pile to track their line of questioning. They can create as many piles as they want, and each can be renamed, duplicated, sorted and filtered.

Piles have a model selection drop-down and 9 different sensemaking tasks implemented as LLM prompts (G3). VisPile breaks down some prompts into adjustable parts to enable customization, including inputs for questions, entity types, and concepts, as well as a slider to adjust the LLM’s creativity. Prompts are always shown verbatim to increase transparency. The prompts were developed through iterative prompt engineering with feedback from our IC collaborators and are available in the supplemental material. They include:

1. “**Analyze** the documents for patterns and insights” – This prompt allows the LLM to apply any techniques to extract evidence.
2. “**Summarize** the main events described” – This prompt attempts to provide a concise description of the main events across the documents.
3. “**Extract** relevant entities (people, locations, etc.)” – This prompt uses the LLM to identify the most important entities across documents, e.g., to align with KG.
4. “**Classify** the relevant topics discussed” – This prompt tries to extract topics across documents to give an overview of what is in them.
5. “**Generate** potential questions that the documents raise” – This prompt helps analysts curate questions the LLM might be able to answer, using the LLM as its own evaluator.
6. “**List** analytic tasks to perform based on the documents” – This prompt gives inspiration for how to parse the documents given their structure.
7. “**Explain** concepts mentioned in the documents” – This prompt helps generate deeper, more focused answers to concepts listed by the user.
8. “**Answer** questions using the documents” – This prompt helps users structure questions for the LLM to answer.
9. “**Custom** prompt with open-ended input” – This prompt is blank; users write whatever they want.

- The KG fact list can also be directly traversed from a pile (G4). The Extract button in piles (Fig. 4A) automatically highlights entities from the KG that appear in LLM-generated text. Clicking on a highlighted entity automatically searches for it in the KG fact list (Fig. 3B) and shows up to 5 related facts. These facts can be traversed by directly clicking on entity names, allowing users to traverse the KG starting with facts related to their pile.

LLM & KG validation features. Several features keep users in the loop of linking LLM- and KG-generated text with source documents, inspiring trust and increasing transparency throughout the sensemaking process.

The Extract button (Fig. 4A) also helps with validation – it can reveal limitations of the LLM like hallucinated terms (G5). The Link button (Fig. 4B) connects each sentence in the LLM response to the most related sentence in documents in a pile using a RAG-like approach. The most similar sentence pairs are underlined and color-coded by the source document. Links can be hovered/clicked to focus the opacity and more easily find pairs. The Suggest button (Fig. 4C)

A Extract Highlights KG entities that appear in LLM-generated text

The screenshot shows the VisPile interface with a pile of documents. A specific sentence from an LLM response is highlighted with red boxes around the words "Kronos guards" and "abila". A large black arrow points from the top left towards this highlighted text. Below the pile, there's a toolbar with buttons for Monospace, EXTRACT, LINK, SUGGEST, and others. The EXTRACT button is highlighted with a red box.

B Link Underlines document sentences most related to LLM sentences

The screenshot shows the VisPile interface with a pile of documents. A specific sentence in a document is underlined with a red line. A large black arrow points from the top left towards this underlined text. Below the pile, there's a toolbar with buttons for Monospace, EXTRACT, LINK, SUGGEST, and others. The LINK button is highlighted with a red box.

C Suggest Adds up to 5 docs from corpus most related to LLM text

The screenshot shows the VisPile interface with a pile of documents. A specific sentence in a document is highlighted with a red box. A large black arrow points from the top left towards this highlighted text. Below the pile, there's a toolbar with buttons for Monospace, EXTRACT, LINK, SUGGEST, and others. The SUGGEST button is highlighted with a red box.

Figure 4. Extract (A), Link (B), and Suggest (C) are buttons in piles that help analysts verify and contextualize evidence from the LLM and KG.

performs another similarity search of the entire LLM response in a pile against the entire document corpus, not just in the pile. The top 5 highest ranked documents are then automatically added to the pile, expanding evidence to corroborate the LLM response. Only documents not already in the current pile out of the top 5 are then added to the pile. This avoids continuously adding documents to the pile that may be less relevant than what is already in the pile. Finally, to help contextualize KG facts, the most connected KG entities, as well as semantically similar entities related to the currently searched term, can be clicked (Fig. 3B) to automatically populate the menu with more facts (**G6**).

3.4. Usage Scenario

To help envision how VisPile supports sensemaking, we analyzed the KRONOS dataset ourselves using VisPile and formatted our findings as insights in a usage scenario. Consider a fictitious seasoned investigative analyst, Bob. Bob has been hearing about a possible kidnapping, and they want to identify relationships between entities in the news for evidence of a collusion ring. Every morning, ≈ 1000 daily news articles flood Bob’s inbox. With limited time to read every document, Bob likes using VisPile to enhance their sensemaking process going from documents to evidence.

Searching for and piling documents. Gaining situational awareness into a document corpus is tricky: What are the documents about? How are they related? Which documents are relevant? To aid Bob in the foraging process, VisPile enhances searching, filtering, grouping, and sorting text documents.

Bob first filters the documents using known keywords like “*kronos*” and groups documents by LLM-generated topic. Then, Bob uses LLM-powered semantic search to rank the relevance of the remaining documents based on open-ended phrases like “*reports related to kidnapping*” and “*missing employees*”, including documents that may not explicitly mention these topics but are relevant (**G1**). If needed, Bob can hover on documents in the documents list or click on the document blocks in a pile to expand the pile and read the source text at any time. Bob then searches for a name, “*edward vann*”, in the KG and finds facts related to police investigations, with source documents that were missed by the LLM semantic search – a potentially related group of documents (**G2**). Once Bob finds relevant documents, they drag them into a pile and rename it to track their line of questioning.

Analyzing document piles. The piling process helped Bob quickly grasp key documents, entities, and events, organizing them into piles for deeper analysis. However, with limited time, Bob cannot read all of the documents in piles to gather evidence. Bob instead leverages the capabilities of LLMs and KGs directly in piles, helping them read less and reason more.

Bob requests a verbose summary using **summarize** to list the main events in their pile and get a lay of the land. They follow up with **answer questions** to extract additional evidence from their pile (**G3**). Customizing LLM tasks helps Bob get more details at the start that they can filter down over time (**G3**). As Bob analyzes events related to the kidnapping, they hit the Extract button to link the LLM response with the KG. A known business leader’s name frequently appears, so Bob clicks

on it to get more evidence from the KG (**G4**). Bob discovers past police run-ins and continues clicking names to explore potential connections (**G4**). They add connected documents to their pile and ask the LLM about the business leader’s financial ties to the police.

Validating LLM & KG results. With a growing set of evidence gathered from the LLM and KG, Bob needs to filter the noise and find signals to gain insight.

Bob first hits the Link button to map the LLM response sentences to source documents in their pile, uncovering new evidence of businesses that were never mentioned when prompting the LLM (**G5**). Wondering if they missed any relevant documents on the financial ties of the businesses, Bob hits the Suggest button and sees several new documents appear that they haven’t explored yet. One mentions an interview, buried years ago and only today just coming to light, of a woman claiming the same businessman paid her hush money not to talk about the disappearance of her daughter (**G5**).

Bob continues exploring connected nodes in the KG to find evidence of relationships with the current suspect and the woman from the suggested documents (**G6**). Several facts deep, the name of a business is listed as semantically related to the woman’s news interview, yet it has not appeared in any KG facts thus far. Curious, Bob searches in the KG for the business, finding several disconnected facts which corroborate the woman’s story and implicate the local business leader (**G6**). With these pieces of verified evidence, Bob alerts their superior that a potential connection has been found. In tandem, the capabilities for validating and contextualizing LLM and KG evidence give Bob confidence that they have identified the right complex relationships hidden in their document corpus.

4. Domain Expert Feedback

Following Sedlmair et al. (2012), we recruited 6 professional analysts from the U.S. Intelligence Community (N1 – 6) to use VisPile and provide domain expert feedback. All analysts have experience analyzing text documents using various visual analytics tools for gathering evidence and maintaining situational awareness. Each analyst spent 60 minutes freely exploring the KRONOS dataset (Whiting et al., 2014). They were encouraged to give verbal feedback on usability and usefulness for supporting unclassified workflows. We manually recorded their think-aloud feedback (Ericsson and Simon, 1984) and inductively collected and organized high-level themes, discussing them amongst all authors (Boyatzis, 1998). This process resulted in emergent feature usage, user workflows, and analytical trade-offs in a task-driven scenario.

Feature usage. The analysts generally liked features in VisPile that they felt could enhance their own sensemaking workflows. Many analysts took advantage of the LLM and KG features to quickly explore alternative hypotheses. N1, 2, 5 used the LLM similarity search for pairwise comparison of document sets. N1 used Link to shortlist new documents to pile and explore, while N3 used topics to pile documents. While “*extract entities*”, “*summarize events*”, and “*analyze documents*” were slightly more commonly used, we observed many LLM tasks being tested. For example, N3 used “*generate questions*” to look for threads to seed their investigation: “*I liked that the GPT can generate things for me. I didn’t have to read 200 documents. I can get a summary or questions to ask about the documents without having to read them.*” The KG was most useful for direct answers to questions, and both N3 and N5 used the KG for this reason. N3 explains that “*If someone wanted me to answer a specific question about the documents, I would go into a knowledge graph to find facts and find related documents.*”

Workflows. Analysts primarily tested features that helped them avoid reading, as N5 summarized: “*There’s a lot of features for not reading documents, and I want to use those features*”. Every analyst started with larger piles and split them apart as they went. N1 explained how LLMs and KGs can enhance domain-specific benefits of this sensemaking loop: “*As an analyst, as I go through large amounts of data, I want to get a large amount of patterns and data that I can’t keep in my head. I tend to start wider because if AI can help me, it helps give an overview much faster and if it hallucinates, I can drill down much faster and validate the information.*” N1 also had a useful approach to staging LLM and KG tasks in this loop: “*Extract entities sounds appealing early on because I can find entity names, summarize for events, classify topics... Later, I want to answer questions, list tasks, etc.*” This may suggest a desire for tools to stage LLM and KG tasks and chain them together at different stages of analysis.

Challenges. Trust and understanding in the LLM and KG was a major concern for every analyst. Many LLM limitations were encountered, such as hallucinating the name for an acronym for N5, giving inconsistent responses for N6, and missing a fact in the dataset for N2 (“*I didn’t know whether the information was in the documents, or if the problem was in the GPT*”). Most had to warm up to trusting the LLM and KG, like N4: “*The trust is never there for me at the start. I take everything with a grain of salt and then read more.*” To overcome these issues, the Link button helped N1, 2, 4 perform fact-checking. N1 had more confidence in using

the LLM for tasks that it may be better at than humans: “*I trust the LLM that it will be able to find information from a large dataset better than I will. I want to start with what the LLM says and go from there.*” N2 used Suggest to help them find the answer to a question that they knew existed in the dataset, but that the LLM was not providing. Traceability for both the KG and LLM are vital to realizing LLM- and KG-infused workflows.

5. Conclusions and Future Work

Using AI to automate and enhance sensemaking can enable new analysis capabilities that take us beyond what is currently possible in existing visual analytics systems. Towards this goal, our work explores design opportunities to infuse LLMs and KGs into visual text analysis. In collaboration with intelligence community experts, we developed VisPile, a visual analytics system for grouping documents, sensemaking, and validating evidence using LLMs and KGs. Feedback from experts highlighted the usefulness of synergizing LLM and KG features, staging LLM and KG operations in a workflow, and using analytic provenance to enhance transparency.

Our work has several limitations. While the KRONOS dataset is widely used, it contains only short, plain-text documents less than 1000 words, making the generalizability of our techniques to other, real-world datasets with longer documents unclear. We only tested OpenAI’s GPT-3.5 Turbo for prompting and KG generation; exploring other LLM sizes and context lengths could impact the quality of insights and the analyst experience. We also did not perform entity matching or KG alignment across people, places, and events, which could enhance KG quality and analytic evidence. Similarly, factual accuracy might improve by exploring UI mechanisms for detecting and correcting errors during KG extraction. Although we explained each feature, the labeling of LLM settings like temperature and the pile validation buttons may have been unclear, confusing users. It is also unclear how limiting KG facts to the top five affected sensemaking, or whether showing all facts would improve understanding. While lists simplify interpretation, the absence of a node-link diagram may have limited analysts’ ability to visualize the overall KG structure. Finally, prior research suggests generative AI may affect expert critical thinking (Lee et al., 2025). For example, analyst feedback provided evidence to suggest that features in VisPile may help in building trust throughout the analysis process. However, more work is needed to study how VisPile affects sensemaking and trust-building across tasks, datasets, and analyst groups.

References

- Andrews, C., Endert, A., & North, C. (2010). Space to think: Large high-resolution displays for sensemaking. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 55–64.
- Baker, J., Jones, D., & Burkman, J. (2009). Using visual representations of data to enhance sensemaking in data exploration tasks. *Journal of the Association for Information Systems*, 10(7), 2.
- Berry, M. W., & Kogan, J. (2010). *Text mining: Applications and theory*. John Wiley & Sons.
- Boyatzis, R. (1998). *Transforming qualitative information: Thematic analysis and code development*. SAGE Publications.
- Cheng, L., Li, X., & Bing, L. (2023, December). Is GPT-4 a good data analyst? In H. Bouamor, J. Pino, & K. Bali (Eds.), *Findings of the association for computational linguistics: Emnlp 2023* (pp. 9496–9514). Association for Computational Linguistics.
- Endert, A., Fiaux, P., & North, C. (2012). Semantic interaction for visual text analytics. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 473–482.
- Endert, A., Hossain, M. S., Ramakrishnan, N., North, C., Fiaux, P., & Andrews, C. (2014). The human is the loop: New directions for visual analytics. *Journal of Intelligent Information Systems*, 43(3), 411–435.
- Ericsson, K. A., & Simon, H. A. (1984). *Protocol analysis: Verbal reports as data*. the MIT Press.
- Grootendorst, M. (2022). Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.
- Hogan, A., et al. (2021). Knowledge graphs. *ACM Comput. Surv.*, 54(4).
- i2 Group. (2024, August). Analyst’s Notebook.
- Kang, Y.-a., & Stasko, J. (2011). Characterizing the intelligence analysis process: Informing visual analytics design through a longitudinal field study. *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 21–30.
- Kim, H., Choi, D., Drake, B., Endert, A., & Park, H. (2019). Topicsifter: Interactive search space reduction through targeted topic modeling. *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 35–45.

- Latif, S., Agarwal, S., Gottschalk, S., Chrosch, C., Feit, F., Jahn, J., Braun, T., Tchenko, Y. C., Demidova, E., & Beck, F. (2021). Visually connecting historical figures through event knowledge graphs. *2021 IEEE Visualization Conference (VIS)*, 156–160.
- Lee, H.-P., et al. (2025). The impact of generative ai on critical thinking: Self-reported reductions in cognitive effort and confidence effects from a survey of knowledge workers. *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*.
- Lewis, P., et al. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474.
- Li, H., et al. (2024a). Knowledge graphs in practice: Characterizing their users, challenges, and visualization opportunities. *IEEE Transactions on Visualization and Computer Graphics*, 30(1), 584–594.
- Li, H., et al. (2024b). Linkq: An llm-assisted visual interface for knowledge graph question-answering. *2024 IEEE Visualization and Visual Analytics (VIS)*, 116–120.
- Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9).
- McColgin, D., Gregory, M., Hetzler, E., & Turner, A. (2006). From question answering to visual exploration. *Proceedings of the ACM SIGIR workshop on Evaluating Exploratory Search Systems, EESS 2006 Workshop*, 47–50.
- ODNI. (2019). *The aim initiative: A strategy for augmenting intelligence using machines*.
- Pan, S., Luo, L., Wang, Y., Chen, C., Wang, J., & Wu, X. (2024). Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7), 3580–3599.
- Pirolli, P., & Card, S. (2005). The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. *Proceedings of international conference on intelligence analysis*, 5, 2–4.
- Sahoo, P., Singh, A. K., Saha, S., Jain, V., Mondal, S., & Chadha, A. (2024). A systematic survey of prompt engineering in large language models: Techniques and applications. *CoRR*, abs/2402.07927.
- Sayler, K. M. (2020). Artificial intelligence and national security. *Congressional Research Service*, 45178.
- Sedlmair, M., Meyer, M., & Munzner, T. (2012). Design study methodology: Reflections from the trenches and the stacks. *IEEE transactions on visualization and computer graphics*, 18(12), 2431–2440.
- Shaib, C., Li, M., Joseph, S., Marshall, I., Li, J. J., & Wallace, B. (2023, July). Summarizing, simplifying, and synthesizing medical evidence using GPT-3 (with varying success). In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of acl 2023 (volume 2: Short papers)* (pp. 1387–1407). Association for Computational Linguistics.
- Shipman, F. M., & Marshall, C. C. (1999). Formality considered harmful: Experiences, emerging themes, and directions on the use of formal representations in interactive systems. *Computer Supported Cooperative Work (CSCW)*, 8(4), 333–352.
- Srivastava, A., et al. (2023). Beyond the imitation game: Quantifying and extrapolating the capabilities of language models [Featured Certification]. *Transactions on Machine Learning Research*.
- Stasko, J., Gorg, C., Liu, Z., & Singhal, K. (2007). Jigsaw: Supporting investigative analysis through interactive visualization. *2007 IEEE Symposium on Visual Analytics Science and Technology*, 131–138.
- Whiting, M., Cook, K., Grinstein, G., Liggett, K., Cooper, M., Fallon, J., & Morin, M. (2014). Vast challenge 2014: The kronos incident. *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 295–300.
- Yan, Y., Hou, Y., Xiao, Y., Zhang, R., & Wang, Q. (2025). Knownet:guided health information seeking from llms via knowledge graph integration. *IEEE Transactions on Visualization and Computer Graphics*, 31(1), 547–557. <https://doi.org/10.1109/TVCG.2024.3456364>
- Zhang, W., Shen, Y., Lu, W., & Zhuang, Y. (2024). Data-copilot: Bridging billions of data and humans with autonomous workflow. *ICLR 2024 Workshop on Large Language Model (LLM) Agents*.
- Zhao, Y., Zhang, Y., Zhang, Y., Zhao, X., Wang, J., Shao, Z., Turkay, C., & Chen, S. (2025). Leva: Using large language models to enhance visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 31(3), 1830–1847.