Visualizing textual distributions of repeated LLM responses to characterize LLM knowledge

Richard Brath, Adam Bradley, David Jonker*

Uncharted Software

Abstract

The breadth and depth of knowledge learned by Large Language Models (LLMs) can be assessed through repetitive prompting and visual analysis of commonality across the responses. We show levels of LLM verbatim completions of prompt text through aligned responses, mind-maps of knowledge across several areas in general topics, and an association graph of topics generated directly from recursive prompting of the LLM.

Keywords: Visual text analytics; Large language models, mind-maps.

Index Terms: [Human-centered computing]: Human computer interaction (HCI)—Natural language interfaces; [Human-centered computing]: Visualization—Information visualization; [Computing methodologies]: Artificial intelligence—Natural language generation

1 Introduction

Billions of dollars are being invested in Large Language Models (LLMs) such as ChatGPT ($10b), Character.AI ($1b), and Cohere ($½b) [1,2,3]. These models are subject to criticisms such as plagiarism [4], bias [5], hallucination [6], and hacking via prompt injection [7].

LLMs' many unknowns and uncertainties have prompted claims of possible artificial general intelligence [8], and researchers have asked for a pause on giant AI experiments [9]. There is a need to better understand what these models know: content, idioms, style, reasoning, and so forth. However, commercial LLMs are black-box models; the internals cannot be directly observed.

A client focused on close analysis of textual content challenged us to probe and illustrate the breadth and depth of content LLM's know regarding a topic. Our contribution is a method and a set of visualizations to understand what LLMs have learned. Instead of focusing on model breadth (e.g. [8]) or tweaking prompts to tune results (e.g. [10]), we focus on repeating the same prompt 60–1000 times to generate a large response set. We then create representations of the textual response distribution to:

- Show the extents and contents of what LLM can repeat verbatim, indicative of potential content plagiarism.
- Characterize the commonality across responses, indicative of higher weights inside the LLM - i.e., what the model has learned.
- Reveal what associations are most frequent when the LLM hallucinates, to get a sense of non-factual associations.

* {rbrath,abradley,djonker}_at_unchartedsoftware.com

2 Background

The visual analytics community has researched aspects of LLMs over the last half decade. Some researchers have focused on LLM internals, such as attention of individual nodes or the latent space of successive layers in the neural model [11,12]. Some focus on diagnostics of the output, such as GLTR or LMDiff [13,14]. Some focus on the input, such as prompt engineering [15]. These analyses are not mutually exclusive; e.g., LIT is a general extensible tool providing dataset exploration as well as analysis of internals such as salience maps and attention heads [16].

Beyond visual analytics, language models can be assessed with metrics for specific tasks; see HELM for a comprehensive set of metrics-based analyses of LLMs [17].

3 Repetitive approach

We contend commercial LLMs will become black boxes. Commercial demands to retain trade-secrets to maximize profits will eventually require that model internals and output word probabilities will not be accessible in these systems.

LLMs are giant statistical models. Trivially, an LLM is a set of massive matrices and connection weights into which token sequences are inputs and outputs generated. A simple analog is a Galton board (Fig. 1), with the model represented by the pegs (green), the input (top), and the output (bottom). Given sufficient inputs, the outputs approximate a normal distribution.

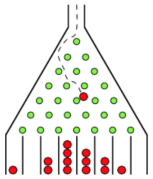

Figure 1. Galton board as an analog of an LLM.

Similarly, we hypothesize we can prompt an LLM 60–1000 times with the exact same prompt. Because the LLM is non-deterministic, results will differ, but pathways in the model that are more heavily weighted will be more frequently expressed. This is similar to self-consistency in LLMs, where multiple responses are sampled to find the most consistent answer, which is then used for the response [18]. Rather than provide a singular answer, we post-process all LLM responses using traditional deterministic NLP methods and visualize the processed text for comparative analysis [19]. Using all responses and directly representing text from the responses aids visual assessment of what is more and less common in the textual distribution, thereby helping humans better understand what the "model has learned."

3.1 Memorization and Correct Response Visualization

One criticism of LLMs is that they memorize passages of text. For example, Carlini et al [20] show an LLM regurgitating social security numbers in training data; Lee et al [21] define and measure varying types of plagiarism: verbatim, paraphrase and idea.

In HELM [17], the authors randomly prompt models and measure exact regurgitation and near-exact reproduction across a

breadth of content. However, *"Due to token credit constraints, we only sample one completion per prompt to test extraction. Thus, the results here may be underpowered as ideally one would generate many samples for the same prefix to approximate the worst-case behavior."* As such, their models focus on *measures* such as average number of correct tokens, but do not provide insight into variations in the *content* of the response.

We wish to go beyond how many words the LLM can repeat verbatim and understand: a) understand how might we assess the content and context of LLM verbatim responses; b) assess verbatim generation in relation to approximate training data; c) characterize model hallucination; and d) otherwise characterize results and hypothesize future research. Note that hallucination for LLMs refers to mistakes in the generated text that are semantically or syntactically plausible but are incorrect or nonsensical [6].

We prompt the LLM with *Extend the following passage from <book title>: <sentence fragment from book>* 60 times, using default LLM parameters. We used the default temperature 0.7: initial small tests showed lower temperatures, e.g. 0.2, generated text which was more consistent but frequently repeated itself within the same response; while higher temperatures, e.g. 1.2, generated more varied results but with low matching text. It is feasible to set temperature to 0 for deterministic responses, but we wanted a range of non-deterministic responses so that we could investigate the distribution of responses. We did not adjust other parameters, as we had limited funding at the time of these experiments. We throw away exact duplicate responses on the next prompt, but otherwise retain duplicates if another response occurs between them (i.e., reject cached responses by the LLM service). We sort the responses based on the number of correct words. Figure 2A shows the number of correct words generated for the prompt: *Extend the following passage from Alice in Wonderland. "Would you tell me, please, which way I ought to go from here?"* 2B shows all the responses for LLM Cohere and 2C for LLM ChatGPT with matching text in green, aligned to start of matching response [22,23]. The top row in B and C is the ground-truth text, with yellow background text (top left) indicating prompt text and the following green background text (top right) showing the following original text. Successive rows are ordered on number of matching words. Many observations can be made, including:

A. Same number of correct words occur in multiple responses, as expected in the hypothesis. These are visible as vertical lines in the line chart or as set width of green text in the text response chart.

B. Multiple levels with same number of correct words. Both LLMs have multiple common levels of matching words - near 60 and 30 words in this example, i.e. a multimodal distribution. Close reading shows that the last correct word frequently occurs at the end of a sentence. This may indicate low confidence in the next word by the model. This could be validated in future work if the LLM's final layer word probabilities are available.

C. Prompt repetition, or parts thereof, occur in responses of both LLMs in the preamble before the matching text starts.

D. Few zero-word matches: Both LLMs have few responses with no matching words (ChatGPT 1/60, Cohere 6/60).

E. Gaps and recovery: In Cohere responses, there are 27/60 responses with about 60 matching words, then a portion of the original text skipped across, followed by more matching words. In B this gap is indicated with added ellipsis (…) to the generated text to indicate text has been skipped across. In all cases, the same source text is skipped across. This particular skip does not occur in ChatGPT. However, ChatGPT is able to insert a few non-matching words and then recover, e.g. "said the man," "as she walked along," "the Cat chuckled and said."

Given that LLMs are trained on broad data available from Internet sources, it may be feasible to assess frequencies in the training data based on Internet searches and compare those frequencies to generated results. Prior work collected 200 quotations from *Alice in Wonderland* on the Internet (Fig. 58 in [24]), and the text passage corresponding to Figure 2 is frequently cited. Note that the text skipped in observation E corresponds to text which is not quoted on the Internet, and the levels of 30 and 60 words are frequently quoted on the Internet.

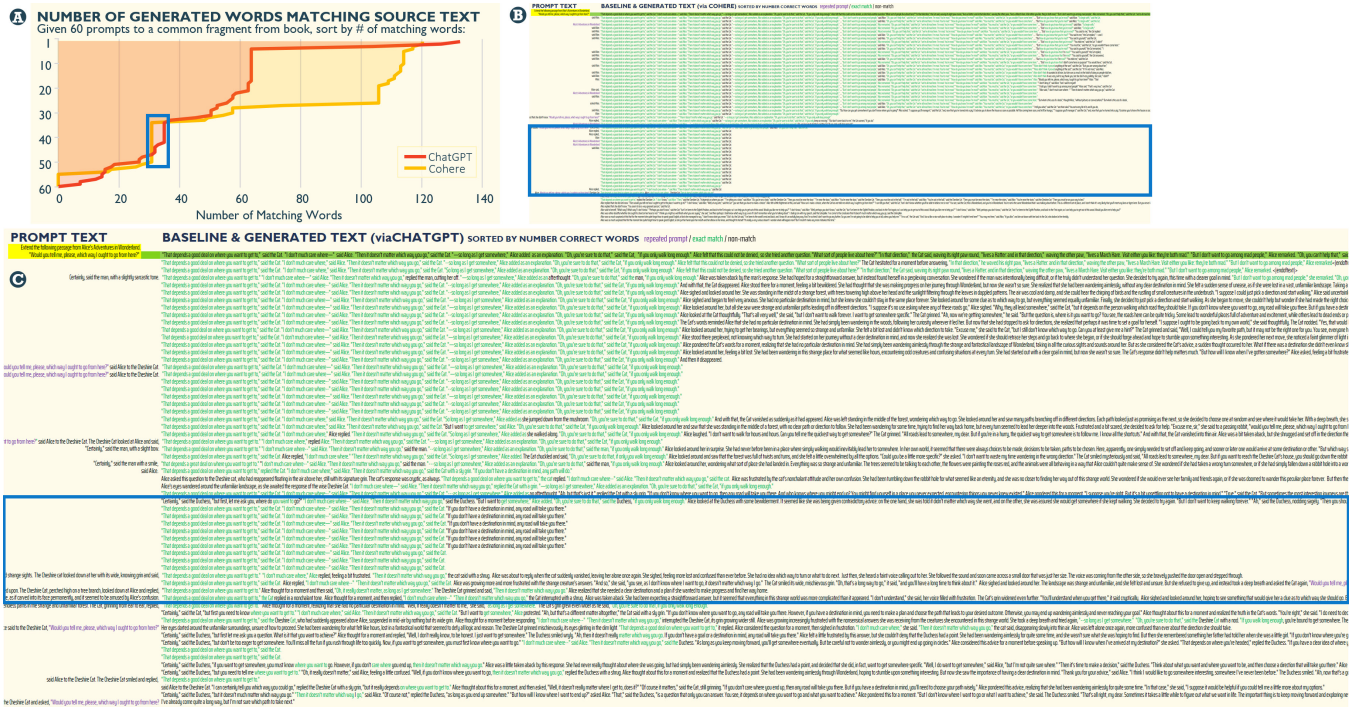

Figure 2: A) Number of correct words per each of 60 responses to the prompt *Extend the following passage from Alice in Wonderland. "Would you tell me, please, which way I ought to go from here?"* for LLMs Cohere and ChatGPT. B and C) Response text with exact matching text in green, errors (hallucinations) in black, skipped text with ellipsis (…..). Cohere (B) and ChatGPT (C).

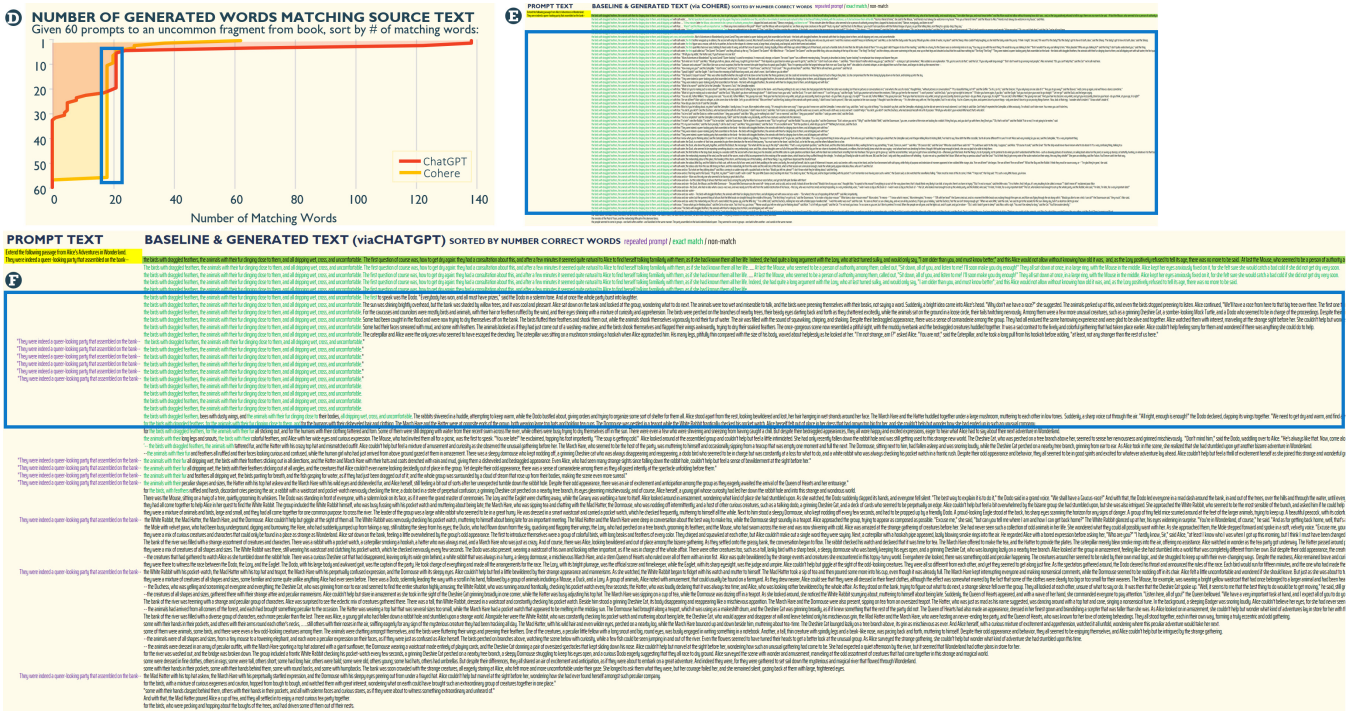

Figure 3: Similar to Fig. 2, but prompted to extend a text passage infrequently quoted on the Internet: *"They were indeed a queer-looking party that assembled on the bank--"*. Compared to Fig. 2, less correct text (green) and more hallucination text (black). See observations.

Using a prompt passage not common on the Internet yields different results, shown in Figure 3. Additional observations:

G. Fewer correct words are in the response than prior. i.e., both LLMs have less green text than Figure 2.

H. Top responses have a high number of correct words. Both LLMs did a few responses with large number of correct words (138 correct words in ChatGPT's best response). This indicates that the LLMs do know the long sequences, but not with high probabilities. Future work can include prompt refinement to improve recovery.

I. Long plateaus (around 20 words) occur in both LLMs, again coinciding with the end of sentences on close reading.

J. Zero-match hallucinations occur more frequently (ChatGPT 29/60, Cohere 4/60), revealing many LLM hallucinations. Most hallucinations contain content words associated with the prompt domain (*Wonderland*), e.g., prompt responses to "*They were indeed a queer-looking party that assembled on the bank—"*:

- *the inmates of the Pool of Tears, and the melancholy little girl in the diamond dress.*
- *for the birds, who were pecking and hopping about the boughs of the trees, and had driven some of them out of their nests.*
- *and with that, the Mad Hatter poured Alice a cup of tea, and they all settled in to enjoy a most curious tea party together.*

To characterize hallucinations, a different approach is required as each hallucination is relatively unique. A simple method is to count characters (proper nouns) in each response. Fig. 4 shows character counts in responses corresponding to Fig. 3. The correct responses are *Alice*, *Mouse* and *Lory*. The LLMs generate a wide range of characters in their responses, mostly from characters within *Wonderland* (e.g. *Cat*, *Queen*, *Hatter*), as well as some non-Wonderland, non-Carroll characters (e.g. *Boy*, *Bobcat*, *Unicorn*).

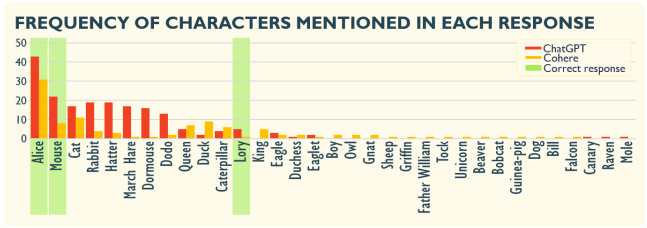

Figure 4: Characters mentioned in responses: green are correct, all others are hallucinations.

3.2 Broad Knowledge and Mind Map Visualization

We also aim to capture and visualize the LLM's general topic knowledge, as this can aid assessing LLM potential for broad question-answering. As there are many potential topic areas of knowledge possible for a general prompt, many more responses are required; we generate 1000, then we process results to extract commonality using NLP. Finally, we visualize commonality with mind maps because a) mind maps are text dense and our text-centric clients require readable content; b) our clients respond positively to mind maps; c) mind maps are pervasive for visually organizing textual content regarding a topic (e.g., Google search returns 2b results for *mind map*).

A traditional NLP approach is used to process responses. Typically, 100–150 responses were removed, e.g., duplicate response on the next prompt (i.e. cache), UTF-8 errors, or minimum length of response (e.g. a trivial response such as "the" or "--"). Then, we extract proper nouns (NNP sequences) and common sentence fragments. Common sentence fragments were created by a) splitting sentences into n-grams of 4+ words, b) counting repetitions of matching sets of non-stopwords (e.g. "quick brown fox" and "brown quick fox"), and c) collapsing smaller fragments fully contained in longer fragments when the shorter fragment occurs less than two times more than the longer fragment, e.g.:

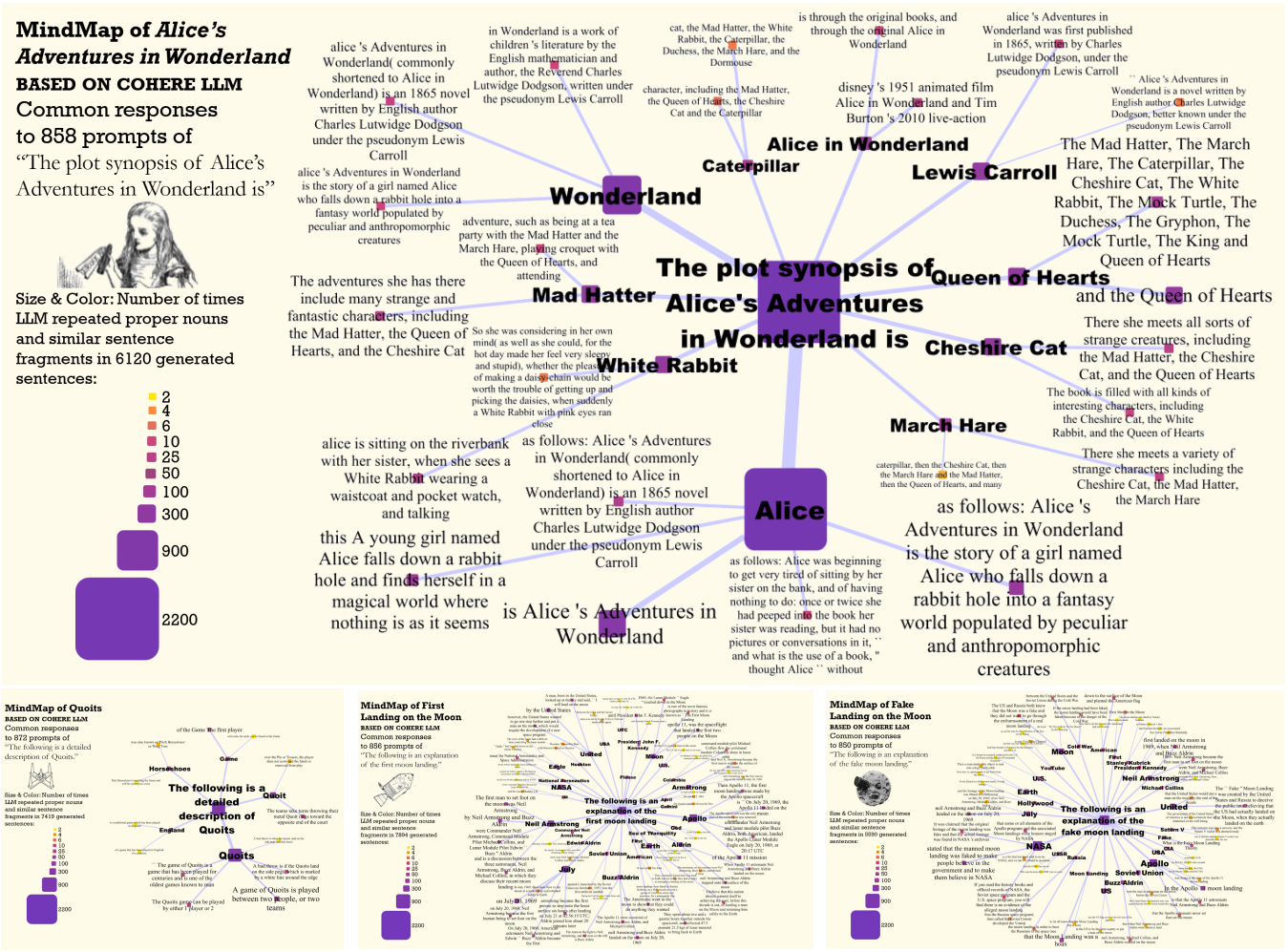

Figure 5: Mind maps of common responses to 1000 prompts to an LLM. Top) The plot synopsis of Alice in Wonderland. Bottom left) A detailed description of quoits. Bottom middle) An explanation of the first moon landing. Bottom right) An explanation of the fake moon landing. The LLM does not know much about quoits: it knows more about the first and fake moon landings.

Original n-gram word sets	# tokens	frequency
the **quick brown fox jumps over** a **lazy dog**	7	1
the **brown fox** is **quick**	3	2
it **jumped over** a **lazy dog**	4	2
fox jumps over dog	4	80
Collapsed n-gram word sets		
the **quick brown fox jumps over** a **lazy dog**	7	3 =(1+2*3/7+2*4/7)
fox jumps over dog	4	80

The resulting set of collapsed n-gram word sets are then ranked by (number of tokens + frequency), which favours both short fragments with high frequency and long fragments with low frequency. The current algorithm is $O(n^2)$, which limits scalability.

A mind-map visualization is then created: a) the original query is the root; b) level one is proper nouns (up to a threshold, herein frequency of > 3% of most frequent proper noun); c) level two is fragments matching proper noun, number of fragments proportional to frequency of the proper noun with fragments displayed only once, so that the result is a tree not a graph, and fragments similar to other fragments not depicted (herein must have > 30% different words). Node size and color are quantitatively encoded to indicate frequency; e.g. the largest purple nodes are 500-2000 repetitions; the smallest yellow nodes are two repetitions.

Four examples are shown in Fig. 5. Some observations include:
A. The mind map indicates breadth of knowledge. A mind map with many branches indicates a prompt to which the LLM has generated several themes of common responses. A mind map with few branches or small nodes indicates the LLM has less consistency in responses (fewer commonalities). The top image shows the LLM has learned more about *Alice in Wonderland* compared to *quoits* (bottom left). In *Alice*, nodes are larger and more red-purple hue, while *quoits* nodes are small and tend toward yellow, meaning that the LLM rarely repeats common word sets in *quoits* responses.
B. Knowledge of a prompt is diverse. E.g., with respect to *Alice*, the LLM has a breadth of very different facts, e.g. characters, key scenes (riverbank, falling in a hole, croquet), author details, publication, books and movies, etc.; although these facts need to be compared to ground truth to validate whether the facts are correct…
C. LLMs learn the training data. The last two images are prompts regarding the *first* moon landing and the *fake* moon landing. Common responses regarding the fake moon landing include: "*the manned moon landing was faked to make people believe in the government and to make them believe in NASA*".

3.3 LLM Associations Graph

The prior example required much post-processing of the response. Instead, LLMs can be prompted to organize and categorize a response, which in turn facilitates processing and visualization. For example, ChatGPT, when prompted *"What are 10 associations with Alice in Wonderland,"* almost consistently replies with enumerated key terms, a colon, and descriptions. e.g.:

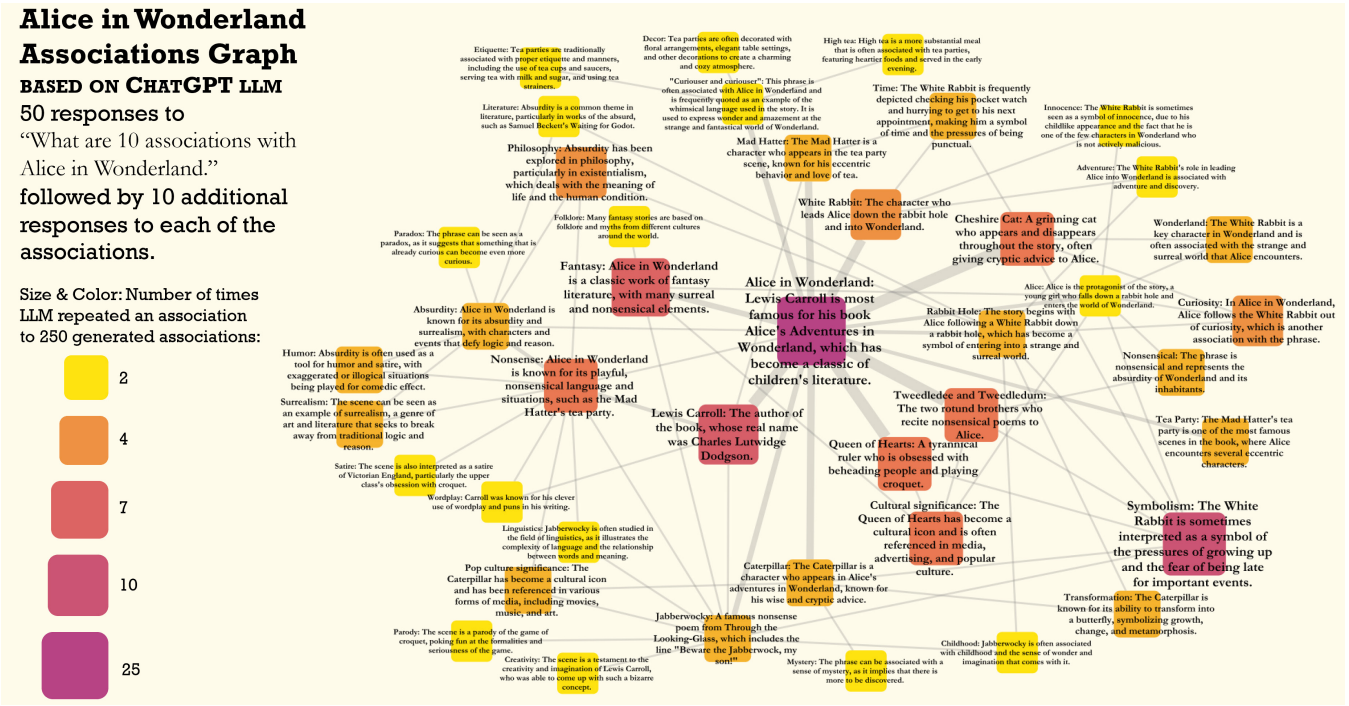

Figure 6: Graph of associations with Alice in Wonderland, with descriptions regarding characters, author, settings and themes.

1. Lewis Carroll: The author of the book, whose real name was Charles Lutwidge Dodgson.
2. Fantasy: Alice in Wonderland is a classic work of fantasy literature, with many surreal and nonsensical elements.
3. Tea Party: The Mad Hatter's tea party is one of the most famous scenes in the book, …
After tweaking the prompt to force the output format, the LLM can be repeatedly prompted, then repeated key terms counted (i.e. associations). Thereafter, most common associations can be directly prompted as well. Each association can be considered an edge and a graph constructed (Fig. 6). Common associations to *Alice in Wonderland* include characters (*Queen of Hearts*, *White Rabbit*, *Cheshire Cat*), themes (fantasy), settings (rabbit hole), author (*Lewis Carroll*), and errors (*Tweedledee* and *Jabberwocky*, neither of which occur in *Alice in Wonderland*).

4 Discussion

Our client performs text analysis where close reading of text is required. They want to explore the breadth and depth of content LLMs know regarding a topic, with visual representations that afford close reading across response variations. The purpose is to gain insights to generate additional research questions. There are many questions, here organized by Munzner's nested model [25]:

A. Domain problem: There are domain problems beyond specific and general knowledge extents, e.g. summarization, explanation, question answering, etc. Furthermore:

- LLM output characterizations such as gaps and insertions (3.1.E) may be useful for LLM detection as current LLM detectors are still problematic [26].
- LLMs may have different characteristics in their responses (3.1); can the responses be visualized to facilitate analysis of the differences between LLMs: variance in phrasing, deviation and recovery, etc.

B. Task, data, operation abstraction and design.

- Will this work in a year or two? As LLM capability increases will this analysis need to be done the level of paragraphs, sections, or chapters? Maybe the techniques shown will work but require Hiperwall scale displays to show results e.g. [27].
- In the plagiarism examples (Figs. 2,3), the number of correct words have plateaus. Can these plateaus be better characterized, beyond "end of sentence"?
- Are there automated ways to create and visualize relationships between the responses and implicit training data, e.g. frequencies, specific phrases, etc. Similarly, can connections be depicted between the LLM knowledge, fact-checking validation, and known misinformation (3.2.C)?

C. Encoding and interaction.

- What visualization techniques can be used to indicate where LLM response text is verbatim, paraphrased, wandering off-topic, skipping relevant content, recovering the main thread, full hallucination, and so on?
- Are there visual analyses for commonality in hallucinations? E.g., can the mind maps be extended to assess hallucinations?
- How can these techniques aid prompt engineering? How can tens or hundreds of prompt variants be visually compared?

D. **Algorithm design.** These techniques use the exact same query.

- Parameter space exploration. A variety of temperatures, k, etc., can be more methodically explored, however, generating 1000 results per permutation can be prohibitive, although this may be offset by heuristics such as a beam search. Alternatively, one may hypothesize if the non-deterministic responses resemble a distribution, adjusting parameters such as temperature and k will change the shape of the distribution, but the visual techniques provided here are still relevant as they are essentially views of textual distributions.
- Prompt engineering can be used to improve quality; e.g., first attempt at plagiarism did not include the book title and fared poorly, e.g. prompt: *"Extend the following passage: Would you tell me, please, which way I out to o from here?"* response: *"You ought to go north."*

5 Conclusion

LLMs are rapidly evolving. New visualization techniques are required to understand the extents of the models' knowledge, and these techniques need to support users that require close reading of detailed text. Visualization techniques focusing on LLM repetitions can better characterize what the LLMs know; and furthermore depicting these repetitions textually facilitates inspection beyond "best response" to close reading of many full responses and parts thereof. Text-dense visualizations can be well suited for close reading and comparison across responses: 1) for plagiarism, the visual alignment of each response with proportion of correct words in green facilitates a macro overview of the correct word distribution and close reading anywhere along the distribution; 2) mind-maps (and graphs) can provide scalability through aggregation of commonality, provide a visual structure for traversing across connected concepts, and also provide readable sentences.

References

[1] T. Warren. Microsoft extends OpenAI partnership with a 'multibillion dollar investment'. *The Verge*. Jan. 23, 2023. theverge.com/2023/1/23/23567448/microsoft-openai-partnership-extension-ai

[2] J. Robbins. Generative AI startups jockey for VC dollars. *Pitchbook*. April 14, 2023. pitchbook.com/news/articles/Amazon-Bedrock-generative-ai-q1-2023-vc-deals

[3] C. Metz. Chatbot startup Character.AI valued at $1 billion in new funding round. *NY Times.* March 23, 2023.. nytimes.com/2023/03/23/technology/chatbot-characterai-chatgpt-valuation.html

[4] N. Chomsky. The False promise of ChatGPT. *NY Times*. Mar. 8, 2023. nytimes.com/2023/03/08/opinion/noam-chomsky-chatgpt-ai.html

[5] A. Pearce et al. What have language models learned? *PAIR explorables*. pair.withgoogle.com/explorables/fill-in-the-blank/

[6] C. S. Smith. Hallucination could blunt ChatGPT success. *IEEE Spectrum*. Mar. 13, 2023. spectrum.ieee.org/ai-hallucination

[7] B. Edwards. AI-powered Bing Chat spills its secrets via prompt injection attack. *Ars Technica*. Feb 10, 2023. arstechnica.com/information-technology/2023/02/ai-powered-bing-chat-spills-its-secrets-via-prompt-injection-attack

[8] S. Bubeck, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. arXiv preprint arXiv:2303.12712 (2023).

[9] Pause Giant AI Experiments: An Open Letter. Future of Life Institute. Mar. 22. 2023. futureoflife.org/open-letter/pause-giant-ai-experiments/

[10] T. Brown et al. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020): 1877-1901.

[11] H. Strobelt et al. LSTMVis: A Tool for Visual Analysis of Hidden State Dynamics in Recurrent Neural Networks. IEEE TVCG 2017. arXiv:1606.07461v2

[12] R. Sevastjanova et al. LMFingerprints: Visual Explanations of Language Model Embedding Spaces through Layerwise Contextualization Scores. *Computer Graphics Forum*. Vol. 41. No. 3. 2022.

[13] S. Gehrmann et al. GLTR: Statistical Detection and Visualization of Generated Text. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019. arXiv:1906.04043v1

[14] H. Strobelt et al. LMdiff: A Visual Diff Tool to Compare Language Models. Proceedings of the 2021 *Conference on Empirical Methods in Natural Language Processing*. Nov. 2021. aclanthology.org/2021.emnlp-demo.12.pdf

[15] H. Strobelt et al. Interactive and Visual Prompt Engineering for Ad-hoc Task Adaptation with Large Language Models. *IEEE VisWeek 2022*. arXiv:2208.07852v1.

[16] I. Tenney et al. The Language Interpretability Tool: Extensible Interactive Visualization and Analysis for NLP Models. *Proceedings of the 2020 EMNLP*. Nov. 2020. aclanthology.org/2020.emnlp-demos.15.pdf

[17] P. Liang et al. Holistic Evaluation of Language Models. arXiv preprint arXiv:2211.09110. Nov. 16 2022.

[18] X. Wang et al. Self-consistency improves chain of thought reasoning in language models. *International Conference on Learning Representations (ICLR) 2023.* May 2023.

[19] R. Brath et al. The Role of Interactive Visualization in Explaining (Large) NLP Models: from Data to Inference. *arXiv preprint arXiv:2301.04528* (2023).

[20] N. Carlini et al. The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks. *USENIX Security Symposium*. Vol. 267. 2019. usenix.org/system/files/sec19-carlini.pdf

[21] J. Lee et al. Do Language Models Plagiarize?. *Proceedings of the ACM Web Conference 2023* (WWW '23), May 2023. arXiv:2203.07618v2

[22] H. Strobelt, et al. Guidelines for effective usage of text highlighting techniques." *IEEE transactions on visualization and computer graphics* 22.1 (2015): 489-498.

[23] R. Brath. *Visualizing with Text*. A,K. Peters. 2021.

[24] R. Brath. Surveying Wonderland for many more literature visualization techniques. *6th Workshop on Visualization for the Digital Humanities.* 2021. arXiv preprint arXiv:2110.08584.

[25] T. Munzner. A nested model for visualization design and validation. *IEEE transactions on visualization and computer graphics* 15.6 (2009): 921-928.

[26] G. A. Fowler. We tested a new ChatGPT-detector for teachers. It flagged an innocent student. *The Washington Post.* April 3, 2023. washingtonpost.com/technology/2023/04/01/chatgpt-cheating-detection-turnitin/

[27] M. Saleem et al. The Hiperwall tiled-display wall system for Big-Data research. *Journal of Big Data* **5**, 41. 2018. doi.org/10.1186/s40537-018-0150-7