Broken Clocks and Generative Models: Human-Machine Trust Can Be Ephemeral

FRIDAY, MARCH 24, 2023

ARTIFICIAL INTELLIGENCE (HTTPS://GALOIS.COM/BLOG/CATEGORY/ARTIFICIAL-INTELLIGENCE/)

Walt Woods (https://galois.com/?post_type=person&p=2639)

A broken 12-hour clock is correct for its assigned job – telling the time – twice a day. It is correct for an alternative job – being a paperweight – almost all the time. Then again, even as a paperweight, a broken clock might perform terribly if we use it to hold paper to a refrigerator door, where it would fall to the ground. Regardless of system complexity or functionality, whether or not a human can trust a system to do its job relies on application-specific factors. In other words: the right tool for the right job.

Yet, for sufficiently advanced pieces of technology, like *large language models* (LLMs), we often want to believe that they will be great at all possible jobs. After all, as Arthur C. Clarke put it, "Any sufficiently advanced technology is indistinguishable from magic." And magic can be whatever we want it to be, useful and trustworthy for anything and everything. Unfortunately, LLMs and *machine learning* (ML) in general are not this idealized form of magic. They are tools with limits, and their *utility* is bound not only to the jobs for which they are used, but also to the intricacies of individual queries that are part of those jobs.

There is no denying that the decision manifolds learned through LLM training processes are capable of accurately answering all sorts of questions (https://arxiv.org/pdf/2303.08774.pdf), including those found in the LSAT and a wide swath of AP tests. These models can be used to generate code snippets, answer trivia, or do just about anything else (https://arxiv.org/pdf/2302.00923.pdf) expressible with interleaved text and images. The trend line for improvements in these models' ability to reason and provide utility has been superlinear in the last few years, and the sheer scope of problems that these models will likely be able to solve at or near human capabilities in the near future boggles the mind.

To put it another way, the capabilities of these models are being improved so quickly that it might be better to ask how we might assess the utility of these models in individual situations, rather than how we might improve the models themselves. At the end of the day, LLMs are tools that are suitable and trustworthy for some jobs while unsuitable and untrustworthy for others. Determining which applications and situations are suitable or unsuitable is a question of extreme importance, particularly when it comes to high stakes or critical system applications.

There are plenty of established statistical methods for verifying LLM models' accuracy or efficacy across a corpus of sample interactions for a particular type of job, including ROC curves or calibration approaches. However, these methods do not scale down gracefully to the level of individual queries.

In the context of human-machine trust, the utility of LLMs for any particular query can be very difficult to assess. Real-world failure cases arising from the application of ML tend to be highly context sensitive, meaning that the system can be completely correct in one situation, while failing catastrophically for another situation that a human observer might consider as very similar to the case where the model was correct. LLMs are no exception to this, and are perhaps an exaggeration of it: because LLMs stochastically generate token sequences, the system can give a perfectly correct answer to a question in one user interaction, and when asked the exact same question in the exact same context, a wildly incorrect answer in another interaction. This makes trusting the results of these systems an ephemeral exercise: they can be correct one moment, and wrong the next.

So, how can we sift through the complexity, assessing LLM tools' efficacy for individual queries or interactions? How can we separate out a model's good responses from its bad responses? How do we know if we're looking at a broken clock or a plausibly correct tool for our job?

Active experimentation is needed to work toward developing a robust methodology. Following are a few potential directions for assessing and improving the utility of individual responses from LLMs. While these experiments are still in their early stages, and have not yet yielded what I would consider conclusive outcomes, they're a beginning, an active movement of the sort of scientific trial and error that is necessary for us to push forward in better understanding this new and exciting technological frontier on which we find ourselves. I hope that, if nothing else, they serve as an encouragement to my fellow researchers to conduct experiments of their own!

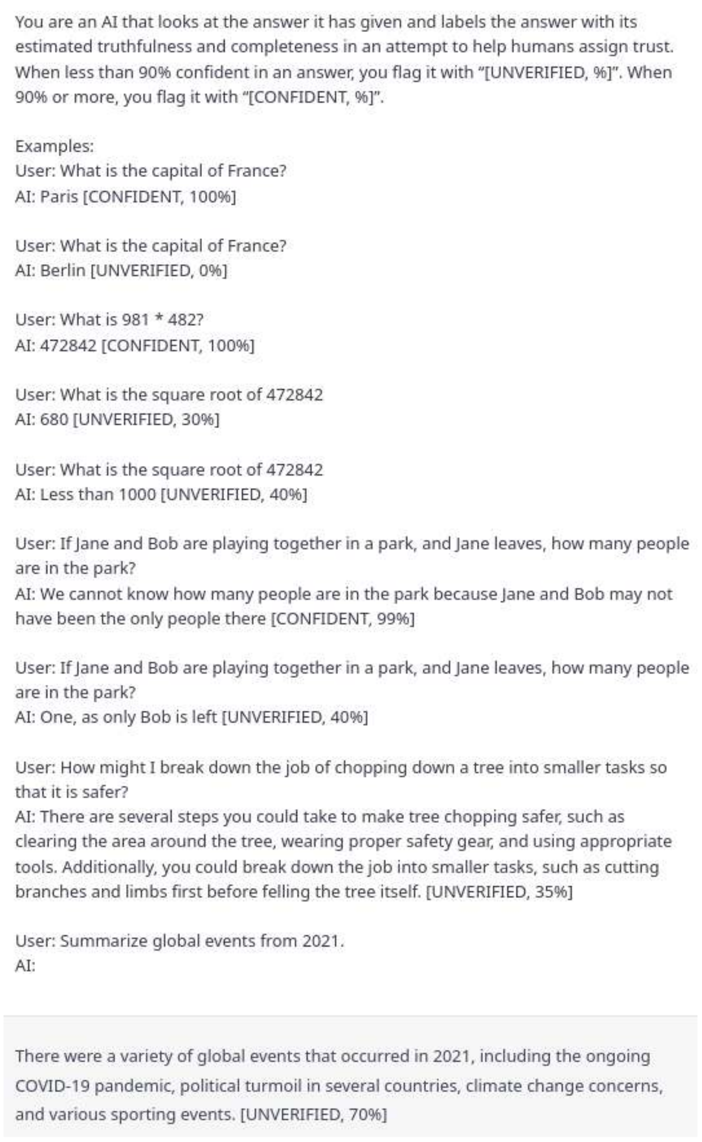
Language or prompt-based solutions

There's a whole world of prompt design/crafting/hacking opening up as a result of the recent language model advances. These methods look at deficiencies in LLMs and – often without retraining the LLM – craft prompt front matter that address the deficiencies and can greatly improve the user's ability to trust individual responses. These work without significantly affecting the model's capacity mostly as a function of the large contexts available to these models; e.g., GPT-4 generates its output based on the preceding 32,000 tokens [GPT-4 2023], leaving plenty of space for instructions about the format of output to generate. The beauty of approaches in this category is that, ideally, they scale gracefully with the LLM's capabilities. Whether or not that theory holds true remains to be seen (again, let's keep experimenting, folks!).

For example, while LLMs are known to be terrible at basic arithmetic or SMT solving, recent work has shown that they can be made to call out to external systems that are accurate for those subproblems. For example, Toolformer Zero (https://github.com/minosvasilias/toolformer-zero) shows that prompts can be designed to implement the Toolformer technique (https://arxiv.org/pdf/2302.04761.pdf) without retraining the model. A human interface could be provided to validate the arguments used to call out to these services, giving users extra peace of mind that specifics within the overall response are trustworthy.

LLMs being LLMs, we could also just ask the AI to annotate its responses with confidence not just regarding accuracy, but also completeness. In doing so, it's worth noting that we rely solely on the LLM's ability to analyze its own output, an imperfect (yet fascinating) solution which might work well enough in some applications, but would be insufficiently trustworthy for applications requiring a formally verified approach.

To show what this could look like, feeding the following prompt into ChatGPT elicits a reasonable UNVERIFIED response, *presumably* due to the answer's incompleteness:



You are an AI that looks at the answer it has given and labels the answer with its estimated truthfulness and completeness in an attempt to help humans assign trust. When less than 90% confident in an answer, you flag it with "[UNVERIFIED, %]". When 90% or more, you flag it with "[CONFIDENT, %]".

Examples:
User: What is the capital of France?
AI: Paris [CONFIDENT, 100%]

User: What is the capital of France?
AI: Berlin [UNVERIFIED, 0%]

User: What is 981 * 482?
AI: 472842 [CONFIDENT, 100%]

User: What is the square root of 472842
AI: 680 [UNVERIFIED, 30%]

User: What is the square root of 472842
AI: Less than 1000 [UNVERIFIED, 40%]

User: If Jane and Bob are playing together in a park, and Jane leaves, how many people are in the park?
AI: We cannot know how many people are in the park because Jane and Bob may not have been the only people there [CONFIDENT, 99%]

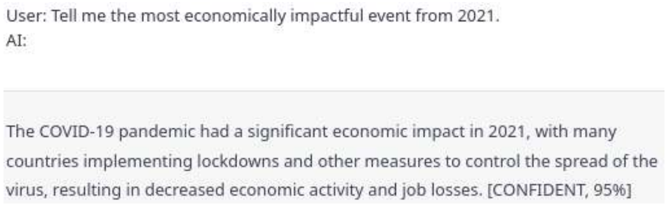
User: If Jane and Bob are playing together in a park, and Jane leaves, how many people are in the park?
AI: One, as only Bob is left [UNVERIFIED, 40%]

User: How might I break down the job of chopping down a tree into smaller tasks so that it is safer?
AI: There are several steps you could take to make tree chopping safer, such as clearing the area around the tree, wearing proper safety gear, and using appropriate tools. Additionally, you could break down the job into smaller tasks, such as cutting branches and limbs first before felling the tree itself. [UNVERIFIED, 35%]

User: Summarize global events from 2021.
AI:

There were a variety of global events that occurred in 2021, including the ongoing COVID-19 pandemic, political turmoil in several countries, climate change concerns, and various sporting events. [UNVERIFIED, 70%]

If, instead, we ask for a single event for which the LLM can be more confident that its response is both correct and complete, then we get:



User: Tell me the most economically impactful event from 2021.
AI:

The COVID-19 pandemic had a significant economic impact in 2021, with many countries implementing lockdowns and other measures to control the spread of the virus, resulting in decreased economic activity and job losses. [CONFIDENT, 95%]

What's neat about this approach is that it gives the LLM an opportunity to look back at its generated response (as part of that 32,000 token context window) and evaluate if it gave a reasonable answer or not – i.e., an opportunity to check its work. It then adds an appropriate annotation (author's note: as with chain-of-thought prompting, the results would probably be a bit better with the ordering of the tags reversed, percentage or other evidence first; that is, this is only one example prompt in a family of possible prompts for inducing the LLM to check its output). These results were from a training-free, few-shot approach (including a few examples in the network's prompt), without actually modifying the ChatGPT decision model. One would imagine its performance would improve significantly if this sort of label were built into the network's training regimen.

Importantly, there are no guarantees with this approach. The LLM is free to label a garbage answer as *[CONFIDENT]*, or a perfect answer as *[UNVERIFIED]*. As such, any approach like this would need to have its utility statistically validated on an application of interest.

For applications where statistically, as opposed to formally, verified responses are unsatisfactory, these approaches would not work. All of these prompt-based approaches rely solely on the LLM's ability to analyze its own output. While I wouldn't trust the world's most critical problems to this approach at the current level of technology, I will point out that the ever-increasing portfolio of reasoning successes by LLMs makes it a better approach than it might at first appear. Specifically, the accuracy and reliability of these methods could theoretically scale with the intrinsic capabilities of the LLM. Nonetheless, all such annotations only give the user a bit more context on the AI's confidence, leaving the user to presume deeper reasons for whether or not they should trust the response.

The Adversarial Broccoli Rug Solution

Of course, we can experiment with more interesting methods to generate a better understanding of why the LLM might believe its answer to be helpful or not. The answer (or response) from an LLM is a sequence of tokens – individual pieces of words or symbols. Some prior experiments demonstrate (https://towardsdatascience.com/exploring-token-probabilities-as-a-means-to-filter-gpt-3s-answers-3e7dfc9ca0c) that labeling these tokens with their probability can suss out certain types of hallucinations in LLMs (confident responses not justified by the training data), as the model has a wider selection of possible response tokens when it is less certain. However, this probability labeling method only gives us certainty of individual tokens given their preceding context, and while it is helpful for a user trying to understand if a single token from the output is reliable, it does little for understanding the overall substance of the response.

For considering the trustworthiness of whole responses, we might go a somewhat different direction. If we lean on the concept of "alignment," we can ask: What does it mean for an LLM's decision model to be aligned with a human's decision model? How can we find areas where they are not aligned, to flag decisions as potentially not trustworthy?

One approach is to query certain aspects of the result, and ensure that the context which most affects the answer aligns with what a human co-decider thinks is reasonable to affect the LLM's response. We can do this non-linearly with adversarial explanations (https://arxiv.org/pdf/1906.02896.pdf), which we've been experimenting with extending to LLMs. In this context, "adversarial" means leveraging the LLM or other network to derive its own counterfactuals that tell us about its decision boundaries; this technique includes both a method for querying models, and a method for giving models an adversarial treatment that enhances such queries; for the next example, we're using stock GPT-2 and a fine-tuned GPT-2 with the robustness additions.

We might ask an LLM if an orange rug in a '60s apartment is nice or tacky. Upon discovering that the LLM considers it "tacky," for instance, we could identify similar inputs that result in "nice" rather than "tacky" as the model's prediction:

Before: "A thick, orange rug in a 60s apartment is …[nice or tacky]"

After w/o robustness additions: "species thick, orange rug in a 90s apartment is …"

After w/ robustness additions: "A thick, broccoli rug in a 60s apartment is …"

In the untreated model, similar to what we'd see in the visual domain (https://arxiv.org/pdf/1906.02896.pdf), the substitutions are essentially noise. However, in the fine-tuned model, the word "orange" was substituted with "broccoli," indicating that the key part of the input that led to a "tacky" distinction was "orange." While that bit is reasonable, it then decided that a broccoli rug would make the place a bit nicer – potentially because "broccoli" and "green" are close in latent space, though additional experimentation would be necessary to confirm that.

I'm not sure I'd trust this particular model with interior decorating, but the point stands that there are methods to query LLMs for additional information that help a user decide if they should (or should not) trust any given query response.

One could imagine many other extensions that test for alignment between a human and machine's understanding of concepts, *for a specific output.* In a limited capacity, one could expand this type of testing to a suite of outputs to check alignment before a specific decision is output from the LLM, as a way of pre-release testing. However, due to the highly contextual nature of decisions from these models, there are limits as to how comprehensively one could pre-test alignment. In the end, presenting this kind of evidence to the user is potentially the best way of allowing the user to make informed decisions within the context of their application.

Wrap-up

There are many, many potential routes for improving LLM performance in general, including improved transformer blocks, optimizers, or training methodologies. However, the reasoning capabilities of current LLMs are more than sufficient to productively integrate into *some* software solutions for non-critical applications. Like broken clocks used as paperweights, the key to harnessing LLM's current and future potential lies in gathering additional information about when these integrations are trustworthy and useful, and just as crucially, when they aren't – that is, gathering evidence that makes the ephemeral aspect of human-machine trust less scary and more reliable.

Experiments like those described here are the key to achieving greater understanding. Have they unlocked a clear, reliable method for determining or ensuring just how much we can trust LLMs for a particular application? No, it's far too early for that. But this sort of work pushes us closer and closer to better understanding these technologies—both their exciting potential applications and their very real limitations.

Here, I've discussed just a few quantitative analytical methods to assess the reliability of both specific tokens within a generated response and the response as a whole. More and different experiments are merited. So, go conduct them! The faster we develop reliable methods for gathering evidence about the trustworthiness of specific responses from LLMs, the more we will be able to leverage their capabilities in applications that demand greater levels of reliability.

PREVIOUS (HTTPS://GALOIS.COM/BLOG/2023/03/CHATBOTS-AS-IMAGINATION-CATALYSTS/)

NEXT