A Platform for the Biomedical Application of Large Language Models

This manuscript (permalink) was automatically generated from biocypher/biochatter-paper@a5297db on October 27, 2023.

Authors

- **Sebastian Lobentanzer** ✉
  0000-0003-3399-6695 · slobentanzer · slobentanzer
  Heidelberg University, Faculty of Medicine and Heidelberg University Hospital, Institute for Computational Biomedicine, Heidelberg, Germany

- **Julio Saez-Rodriguez** ✉
  0000-0002-8552-8976 · saezrodriguez · saezlab
  Heidelberg University, Faculty of Medicine and Heidelberg University Hospital, Institute for Computational Biomedicine, Heidelberg, Germany

✉ — Correspondence possible via GitHub Issues or email to Sebastian Lobentanzer <sebastian.lobentanzer@gmail.com>, Julio Saez-Rodriguez <pub.saez@uni-heidelberg.de>.

Abstract

The wealth of knowledge we have amassed in the context of biomedical science has grown exponentially in the last decades. Consequently, understanding and contextualising scientific results has become increasingly difficult for any single individual. In contrast, current Large Language Models (LLMs) can remember an enormous amount of information, but have notable shortcomings, such as a lack of generalised awareness, logical deficits, and a propensity to hallucinate. To improve biomedical analyses, we propose to combine human ingenuity and machine memory by means of an open and modular conversational platform, biochatter, exemplified in the web application ChatGSE. We safeguard against common LLM shortcomings using general and biomedicine-specific measures and allow automated integration of popular bioinformatics methods. Ultimately, we aim to improve the AI-readiness of biomedicine and make LLMs more useful and trustworthy in research applications.

Main

Despite our technological advances, biology and biomedicine continue to pose incredible challenges [1]. We measure more and more data points with ever-increasing resolution to such a degree that their analysis and interpretation have become the bottleneck for their exploitation. One reason for this challenge may be the inherent limitation of human knowledge [2]. Even seasoned domain experts cannot know the implications of every molecule, be it metabolite, DNA, RNA, or protein, even in their own domain. In addition, biological events are context-dependent, for instance with respect to a cell type or specific disease.

Large Language Models (LLMs) of the current generation, on the other hand, can access enormous amounts of knowledge, encoded (incomprehensibly) in their billions of parameters [5]. Trained correctly, they can recall and combine virtually limitless knowledge from their training set. ChatGPT has taken the world by storm, and many biomedical researchers already use LLMs in their daily work, for general as well as bioinformatics-specific tasks [7]. However, the current, predominantly manual, way of interacting with LLMs is virtually non-reproducible, and their behaviour can be erratic. For instance, they are known to hallucinate: they make up facts as they go along, and, to make matters worse, are convinced - and convincing - regarding the truth of their hallucinations [7]. While current efforts towards AGI (Artificial General Intelligence) manage to ameliorate some of the shortcomings by ensembling multiple models [9] with long-term memory stores [10], the current generation of AI does not inspire adequate trust to be applied to biomedical problems without supervision [8]. Additionally, biomedicine demands greater care in data privacy, licensing, and transparency than most other real-world issues.

A major aim of computational biology is to distil high-dimensional molecular measurements into a humanly digestible form by projecting the measurements into a lower-dimensional space composed of gene programs, pathways, or other functional groupings of biological entities, for example via gene set enrichment analyses. However, even this distilled knowledge requires advanced expertise and thorough literature research to effectively interpret and exploit, and benchmarking the methods' performance is non-trivial [11].

To improve and accelerate this interpretation and exploration, we have developed biochatter, a platform for communicating with LLMs specifically tuned to biomedical research, the use of which we demonstrate in a conversational web interface, ChatGSE (Figure 1). The platform guides the human researcher intuitively through the interaction with the model, while counteracting the problematic behaviours of the LLM. Since the interaction is mainly based on plain text, it can be used by virtually any researcher. We engineer prompts around the queries of the user to improve model performance

with regard to biomedicine, and automate the integration of popular bioinformatics methods, such as differential expression and gene set enrichment (Supplementary Note Prompt Engineering).

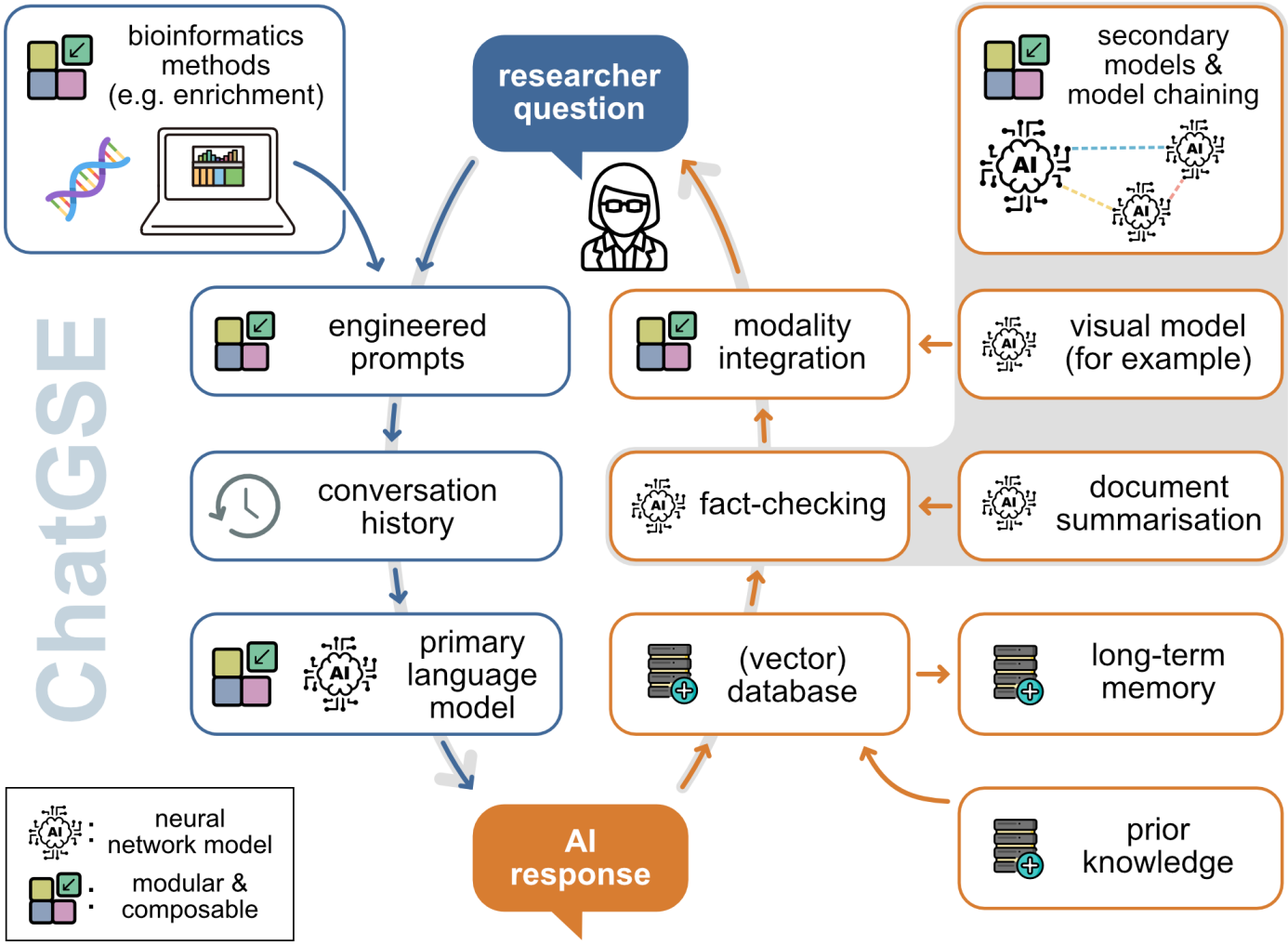

Figure 1: The ChatGSE composable platform architecture (simplified). The user submits a question about a topic of interest (e.g., an experiment) along with the low-dimensional results of a bioinformatics analysis (top left). The platform's main response circuit (blue) composes a number of specifically engineered prompts and passes them (and a conversation history) to the primary LLM, which generates a response for the user based on all inputs. This response is simultaneously used to prompt the secondary circuit (orange), which fulfils auxiliary tasks to complement the primary response. In particular, using search, the secondary circuit queries a database as prior knowledge repository and compares annotations to the primary response. The knowledge graph can also serve as long-term memory extension of the model. Further, an independent LLM receives the primary response for fact-checking, which can be supplemented with context-specific information by a document summarisation model. If this "second opinion" differs from the primary response, a warning is issued. The platform is composable in all aspects, in principle allowing arbitrary extensions to other, specialised models for additional tasks orchestrated by the primary LLM.

On the model side, we implement several measures in addition to the prompt engineering around the user's queries. For instance, we deploy a second model to safeguard the factual correctness of the primary LLM's responses (Supplementary Note Correcting Agent). These interactions are handled by a pre-programmed conversational "Assistant," which dynamically orchestrates LLM agents with distinct tasks using a Python model chaining framework [9]. Using vector database approaches, the user's prompts can be further supplemented with information extracted from pertinent, user-provided literature (Supplementary Note In-context Learning).

References

1. **Study reveals cancer's 'infinite' ability to evolve**
   BBC News
   (2023-04-12) https://www.bbc.com/news/health-65252510

2. **Capacity limits of information processing in the brain**
   René Marois, Jason Ivanoff
   Trends in Cognitive Sciences (2005-06) https://doi.org/d5gmqt
   DOI: 10.1016/j.tics.2005.04.010 · PMID: 15925809

3. **PaLM: Scaling Language Modeling with Pathways**
   Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, … Noah Fiedel
   arXiv (2022) https://doi.org/kfxf
   DOI: 10.48550/arxiv.2204.02311

4. **LaMDA: Language Models for Dialog Applications**
   Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, … Quoc Le
   arXiv (2022) https://doi.org/kmfc
   DOI: 10.48550/arxiv.2201.08239

5. **GPT-4 Technical Report**
   OpenAI
   arXiv (2023) https://doi.org/grx4cb
   DOI: 10.48550/arxiv.2303.08774

6. **Reference-free and cost-effective automated cell type annotation with GPT-4 in single-cell RNA-seq analysis**
   Wenpin Hou, Zhicheng Ji
   Cold Spring Harbor Laboratory (2023-04-21) https://doi.org/gsznzg
   DOI: 10.1101/2023.04.16.537094 · PMID: 37131626 · PMCID: PMC10153208

7. **How will generative AI disrupt data science in drug discovery?**
   Jean-Philippe Vert
   Nature Biotechnology (2023-05-08) https://doi.org/gsznzd
   DOI: 10.1038/s41587-023-01789-6 · PMID: 37156917

8. **Foundation models for generalist medical artificial intelligence**
   Michael Moor, Oishi Banerjee, Zahra Shakeri Hossein Abad, Harlan M Krumholz, Jure Leskovec, Eric J Topol, Pranav Rajpurkar
   Nature (2023-04-12) https://doi.org/gr4td4
   DOI: 10.1038/s41586-023-05881-4 · PMID: 37045921

9. **Langchain** https://python.langchain.com/

10. **AutoGPT Official**
    AutoGPT Official
    (2023-09-18) https://autogpt.net/

11. **Toward a gold standard for benchmarking gene set enrichment analysis**
    Ludwig Geistlinger, Gergely Csaba, Mara Santarelli, Marcel Ramos, Lucas Schiffer, Nitesh Turaga, Charity Law, Sean Davis, Vincent Carey, Martin Morgan, … Levi Waldron

Briefings in Bioinformatics (2020-03-09) https://doi.org/ggs7tp
DOI: 10.1093/bib/bbz158 · PMID: 32026945 · PMCID: PMC7820859

12. **The TRUST Principles for digital repositories**
Dawei Lin, Jonathan Crabtree, Ingrid Dillo, Robert R Downs, Rorie Edmunds, David Giaretta, Marisa De Giusti, Hervé L'Hours, Wim Hugo, Reyna Jenkyns, … John Westbrook
Scientific Data (2020-05-14) https://doi.org/ggwrtj
DOI: 10.1038/s41597-020-0486-7 · PMID: 32409645 · PMCID: PMC7224370

13. **Large Language Models are Zero-Shot Reasoners**
Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, Yusuke Iwasawa
arXiv (2022) https://doi.org/gr263v
DOI: 10.48550/arxiv.2205.11916

14. **HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in Hugging Face**
Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, Yueting Zhuang
arXiv (2023) https://doi.org/gskd97
DOI: 10.48550/arxiv.2303.17580

Supplementary Materials

In our Supplementary Notes, we explain the functions of our platform in more detail. Please note that several of the mentioned features, particularly more advanced ones, are in early developmental stages. For practical reasons, we divide our software into two distinct packages, the user interface (UI) component ChatGSE and the Python backend library biochatter. This also reflects how we expect the platform to be used: as a generic backend library for the development of custom UIs. For an up-to-date overview and preview of current functionality of the platform, please visit the online preview. Both libraries are developed in Python (version 3.10), according to modern standards of software development [12]. We use Streamlit (version 1.21.0, https://streamlit.io) for the web UI. We include a code of conduct and contributor guidelines to offer accessibility and inclusivity to all that are interested in contributing to the framework.

Prompt Engineering

Recent experience with Large Language Models (LLMs) shows that the clever engineering of model prompts can yield drastic performance increases. For example, when performing logical inference, simply adding "Let's think step by step" to the end of a question prompt increased LLM performance from ~20% to ~80% [13]. As such, we highly prioritise the identification and adjustment of prompts tuned exactly to the requirements of the specific task to be performed, respecting "general" rules of interacting with LLMs as well as biomedicine-specific issues.

We designed the backend of ChatGSE to be completely flexible with regard to the application and rearrangement of prompts and prompt templates (which allow the insertion of variables such as the user input question or data). For templating, we use the corresponding generic functionality provided by LangChain [9], while adding our biomedicine-specific layer on top. We provide general prompts to the primary model, setting it up to be helpful and concise in its responses, and individual prompts for each tool we want the primary model to "understand." This includes explanation of the method itself as well as structural information about the data file containing the results.

To make this functionality available to all users, we provide a "Prompt Engineering" tab in the ChatGSE application, which allows the modification (or removal) of existing prompts, as well as the addition of new ones. We also provide functionality to import and export these prompts in JSON format to facilitate reproducible and shareable biomedical prompt engineering. To discuss and share examples

of useful or ineffective prompts, we encourage all users to join the #chatgse stream in our freely accessible Zulip channel at https://biocypher.zulipchat.com, or the GitHub discussion thread at https://github.com/biocypher/ChatGSE/discussions/11.

To further improve reproducible prompt engineering, we will establish online datasets for benchmarking LLM performance on biomedical tasks. These datasets will be available through the ChatGSE frontend and the performance of specific prompt sets on these benchmarks will be captured using an online leaderboard system. This way, the most useful prompt sets for specific tasks can be maintained concurrently with the development of the LLMs.

Correcting Agent

The propensity of LLMs to hallucinate untrue facts necessitates control mechanisms and guardrails for their application in research, particularly in high-stakes fields such as biomedicine [7]. Some corrective incentive can be included in the instructions to the primary model, for instance by including a "Criticism" directive such as "Constructively self-criticise your big-picture behaviour constantly." However, this does not guarantee to prevent hallucinations completely, as the same model that hallucinates also is responsible for correction.

Thus, we implement a modular combination of primary and corrective model, where the corrective agent – a "second opinion" – is set up with instructions tuned exactly to the task of fact-checking the responses of the primary model ("You are a fact-checker. Please judge the following statement for its factual correctness. ..."). The performance of the corrective agent can be further increased by using a more powerful model (e.g., moving from gpt-3.5-turbo to gpt-4), by pre-processing the primary model's response (e.g., splitting the response into single sentences and fact-checking each sentence individually), and comparing the primary model's statements to prior knowledge stored in a knowledge graph connected to the chat platform. As model development advances and other parties create models as powerful as OpenAI's, we foresee it will be useful to combine models from different suppliers to increase diversity between primary and corrective models. Using ChatGSE's modular framework, arbitrary numbers of corrective models can be added to the LLM chain.

To allow all users to interact with correcting functionality, we include a dedicated "Corrective Agent" tab for adjusting settings of the corrective model independent of the settings for the primary model. We also facilitate the testing of corrective agents and their prompts by providing a free-text field for sending false information to the corrective agent. This is necessary since some models, particularly those from OpenAI, are heavily regulated to not purposely provide false information, even for testing purposes.

The comparative power of the LLM agents can be further increased by connecting to a vector database containing embeddings of the contents of specific user-supplied documents. For more information, see the following Supplementary Note 3: In-context Learning.

In-context Learning

While the general knowledge of current LLMs is extensive, they may not know how to prioritise very specific scientific results, or they may not have had access to some research articles in their training data (e.g., due to their recency or licensing issues). To bridge this gap, we can provide additional information from relevant publications to the model via the prompt. However, we cannot add entire publications to the prompt, since the input length of current models still is restricted; we need to isolate the information that is specifically relevant to the question given by the user. To find this information, we perform a similarity search between the user's question and the contents of user-

provided scientific articles (or other texts). The most efficient way to do this mapping is by using a vector database.

The contextual background information provided by the user (e.g., by uploading a scientific article of prior work related to the experiment to be interpreted) is split into pieces suitable to be digested by the LLM, which are individually embedded by the model. These embeddings (represented by vectors) are used to store the text fragments in a vector database; the storage as vectors allows fast and efficient retrieval of similar entities via the comparison of individual vectors. For example, the two sentences “Amyloid beta levels are associated with Alzheimer’s Disease stage.” and “One of the most important clinical markers of AD progression is the amount of deposited A-beta 42.” would be closely associated in a vector database (given the embedding model is of sufficient quality, i.e., similar to GPT-3 or better), while traditional text-based similarity metrics probably would not identify them as highly similar.

By comparing the user’s question to prior knowledge in the vector database, we can extract the relevant pieces of information from the entire background. These pieces (for instance, single sentences directly related to the topic of the question) are then sufficiently small to be directly added to the prompt. In this way, the model can learn from additional context without the need for retraining or fine-tuning. This method is sometimes described as in-context learning [14].

To provide access to this functionality in ChatGSE, we add a “Document Summarisation” tab to the platform that allows the upload of text documents to be added to a vector database, which then can be queried to add contextual information to the prompt sent to the primary model. This contextual information is transparently displayed in the main chat window. Since this functionality requires a connection to a vector database system, we provide modular connectivity to several standard vector database providers, such as Pinecone, Weaviate, or Milvus.