Paper ML14

# Leveraging NLP for Clinical Trial Efficiency: Development of a Protocol-Aware AI Chatbot

Sairam Tangirala[a], Seamus Gallivan[b], Jagan Mohan Achi[c],
Jazz Pharmaceuticals, Palo Alto, USA

## ABSTRACT

Jazz Pharmaceutical PLC's technological transformation is modernizing its operations by integrating AI/ML tools to streamline clinical trial workflows. Its Integrated Data Analytics and Statistical Programming (IDASP) division has developed a proof-of-concept (PoC) AI-powered chat agent to improve clinical protocol document management by enabling natural language interactions. This chat agent leverages Retrieval-Augmented Generation (RAG) techniques to enhance information retrieval while mitigating LLM hallucinations, ensuring accuracy, compliance, and efficiency. This GenAI tool was tested for its performance using a structured questions-based assessment tool, for evaluating its effectiveness in providing contextually accurate responses. The PoC demonstrates the potential for AI-driven solutions in the pharmaceutical industry while highlighting the need for continuous refinement, regulatory oversight, and human-in-the-loop mechanisms.

## INTRODUCTION

Jazz Pharmaceuticals is a global biopharmaceutical company undergoing a technological transformation to modernize its operations and to adopt cutting-edge AI/ML technologies in its operations. The organization is making progress on its technology-upgrade roadmap that includes leveraging Generative AI and machine learning tools to streamline and automate its clinical trials workflow pipeline. In the clinical trials industry, the management of clinical trials, the involved data analysis, document processing, audit and reporting requirements are human resource intensive processes, and availability of Generative AI tools presents an opportunity to automate routine and 'low intelligence' tasks, to create innovative workflow solutions, and to harness recently available AI/ML technology efficiencies. The technological goal is to explore, develop and integrate AI/ML solutions for driving innovations while maintaining oversight, ensuring compliance, fostering inter-team collaborations, and supporting business goals of the organization.

Clinical trials result in the creation of extensive documentation consisting of brochures, informed consent forms, statistical analysis protocols, case report forms, regulatory submission documents, and other materials related to the trial. There are additional logistics overhead of managing version controls, saving audit logs and doing it manually may lead to inconsistencies, outdated references and compliance risks. Another challenge is the presence of clinical-trials data in multiple data-silos, and structured/unstructured formats, this makes manual data retrieval search, retrieval, and analysis further time-consuming, laborious, and inefficient. These manual tasks are often time-consuming as the data is both multi-formatted, unstructured and in multiple data silos.

The above-mentioned document processing workflows are a great test-case for AI-powered systems that can contextually mine and index data that includes multiple forms such as text, images, figures, tables, listings, media recordings, etc. The IDASP division at Jazz Pharmaceuticals has identified several such use-cases to explore usage of AI agents. This white paper discusses the design, implementation and testing of a chat-agent that accesses internal proprietary resources interacts with business users to provide answers to user's natural language queries based on specific clinical study protocol(s). The IDASP's leadership believes that such a Generative AI chat-agent can significantly enhance business user's experience in interacting with clinical trial protocol(s) by providing business and clinical trials related technical information more accessible, efficient, and accurate. The human in the loop requirement is supported by the chat-agent by providing references to internal document sources that are used in producing chat-agent's response. Clinical trials related protocol documents are often verbose and contain technical information, and a chat-agent can enable business users to ask specific questions (using natural language) and receive concise, accurate responses instantly. This makes the business workflow efficient by reducing the need to manually identify and search through (often multiple) protocol documents. The chat-agent provides concise, accurate responses based on the most up-to-date protocol version, minimizes misinterpretation, ensures consistency, and provides references to manually verify the responses. Additionally, it supports cross-functional collaboration by offering clear explanations tailored to varying expertise levels among users. It also enables exploratory queries by helping users explore "what-if" scenarios and to better understand protocol's implications in the trials.

The authors believe that traditional clinical trial document management systems can be upgraded with AI-augmented tools to keep pace with the scale and inherent complexities (often leading to increased operational costs and potential delays in drug development) and the contemporary state-of-the-art AI/ML tools may provide opportunities

[a] ST works as a principal data scientist

[b] SG works as a senior data scientist

[c] JA works as the head of data analytics and statistical programming

to reduce inefficiencies, compliance risks, security vulnerabilities, and information retrieval challenges. Here are several examples of organizations using AI/ML technology to boost their workflows. There are several use cases where AI-based innovations have been reported in the clinical-trials industry, some of them include [1, 2, 3, 4, 5].

**Chat-Agent (Proof-of-Concept)**

We present some of the interesting cases that Jazz Pharmaceuticals is considering in its technological upgrade roadmap.

- Assisting in drafting clinical trial protocols by analyzing historical studies and regulatory guidelines.
- Chat-agents can provide business users with trial-specific information, answer FAQs, and guide them in their business case.
- Help in reviewing and compiling regulatory documents for trial-related submissions.

This AI/ML exploration PoC project at IDASP involved designing, developing, and testing a PoC GenAI application that demonstrates GenAI capabilities to business users with acceptable sensitivity and specificity to proprietary data and knowledge base. In this use-case, we did not fine-tune and/or train the LLM model (defined below) as it is a very computationally and time intensive endeavor. Additionally, the limitation of ground truth knowledge available for this use-case also prevented fine-tuning the foundational LLM during the PoC creation.

A Large Language Model (LLM) is a machine learning model which has been designed by using self-supervised learning [6] and training on existing vast corpora of natural language text extracted from various sources on the internet including clinical-trials domain. Typical use-cases of using LLMs include applications requiring natural language text-understanding and generating natural text. At the time of authoring this document, there were several LLMs accessible to users and were used in clinical trials. Usually the LLMs (also know as foundational models) are based on Generative Pre-trained Transformer (GPT) architecture [7, 8, 9]. A few examples include GPT-3.5 and GPT-4, LLAMA Series (7B, 13B, and 70B) [10], TrialGPT [11] and PubMedBERT [12]. In the GPT framework, the LLMs are created using artificial neural networks, which have been pre-trained on a very large corpus of natural language text. As the LLMs are trained on broad-topics data at scale, they are readily not useful in understanding and responding accurately to a specific context the use-case. For example, LLMs perform poorly in understanding and responding correctly to prompts/questions that are based on proprietary/private information because the model has no exposure to text information which was not available during its training phase. In such situations, LLMs based GenAI agents have been observed to hallucinate and respond to user prompts/questions in a general way and sometimes in ways that seem to make sense (in relation to the context) but are not accurate. Such response of LLMs is termed as “hallucination” [13,14, 15].

During hallucinations, LLMs produce text outputs that are grammatically correct and seem reasonable to the user but are factually incorrect or nonsensical in relation to the context that the use-case in context. “Hallucinations” in this context means the generation of false or misleading information that seems to be accurate in general context and inaccurate in specific context. These hallucinations are thought to occur due to various factors, such as limited training data, biases in the model, or the complexity of natural language. LLM’s hallucinations are very concerning in use-cases that require high levels of accuracy and where LLMs responses may have a significant impact on domains like clinical trials, healthcare, legal matters, or engineering. Therefore, in clinical-trials use-cases, it is essential not to rely on out-of-the box responses of LLMs and to design AI-agents that have humans in-the-loop and have been exposed to (or have access to) proprietary restricted information. Without being exposed to (or not having access to) the ground truth knowledge, LLMs should not be used.

There are several strategies to reduce hallucinations by exposing LLMs to the ground truth, appropriate to the use-case. The treatment of strategies to reduce LLM’s hallucinations is beyond the scope of this article.

**Method**

During development of this PoC chat-agent, the authors prioritized data privacy and intellectual property (IP) protection, while working with GenAI foundational models. For this, we implemented strict data governance frameworks, adhering to appropriate regulations to safeguard sensitive clinical trial data, patient information, and proprietary research findings. Given the data-exposure risks associated with GenAI models, the authors, in consultation with the Information Security team and the leadership’s advice, were very cautious in the approach. We worked with publicly available protocols in this project and worked within a controlled isolated environment, using on-premises infrastructure not accessible to the outside world.

The chat agent was specifically designed for handling business user queries pertaining to very specific documents. We used the RAG technique [16, 17, 18] to have the chat agent generate natural language responses to user’s natural language queries. The RAG approach is a way to provide LLMs an access to restricted knowledge by creating and making contextual information accessible to them. The contextual data is encoded, indexed, and stored separately from the LLM. LLM has no direct access to the original context documents and has controlled access to encoded indexed data stored in a different location (than the LLM). In summary, RAG helps LLMs to be more

knowledgeable by using the stored contextual information to answer questions better. This approach enables the chat agent to generate accurate, contextually rich, and domain-specific answers while mitigating LLM hallucinations.

In this section, we briefly discuss the general steps used in developing the chat agent.

- Chunking the input document: The LLMs cannot process large volumes of texts at once. So, we split the input context document into smaller, meaningful text chunks before embedding (explained below). We used the simplest chunking technique called fixed-size chunking to split the natural language text in the input document into fixed sized tokens (512 token size chunks).
- Creating embeddings: Each chunk was converted into a vector representation (also known as vector embeddings/embeddings) using a pre-trained embedding model. These embeddings help in retrieving relevant information efficiently. In the PoC, we used the embedding model "all-mpnet-base-v2". It is a powerful transformer-based model known for its strength in capturing sentence-level meaning with high accuracy. It is able to create dense embeddings that effectively represent semantic nuances and relationships within text. This makes this embedding model suitable for applications requiring deep contextual understanding (needed in the chat agent PoC). We also used the "paraphrase-MiniLM-L6-v2" and "all-MiniLM-L6-v2" embedding models which are lightweight (in memory usage) models and are optimized for generating compact and efficient embeddings. These are suited for tasks like paraphrase detection, where identifying semantic similarity is crucial, and in infrastructures with computational constraints. Their small size and reduced computational overhead make them excellent for real-time applications while maintaining reasonable performance in encoding semantic information
- Creating and storing an index of contextual data: We created a FAISS (Facebook AI Similarity Search) index to store the vector embeddings for fast and scalable retrieval. The steps include:
    - Converting document chunks into embedding vectors
    - Indexing these vectors in FAISS for efficient nearest neighbor search
    - Storing metadata (e.g., document IDs, chunk position) separately
- Query processing and contextual data retrieval: When a business user inputs a query, the natural language query is converted into an embedding vector using the same embedding model (chosen during embedding of the document chunks). Then, FAISS performs a similarity search to retrieve the most relevant document chunks. The retrieved chunks are then passed (as additional context) to the LLM for response generation.
- Using LangChain for LLM integration: We used LangChain package from Python to integrate retrieved embeddings with LLMs. The key components of LangChain used in the PoC included:
    - FAISSRetriever: for retrieving relevant chunks from FAISS vector store
    - LLMChain: for connects the retrieved vectors with the LLM for contextual response generation
    - Memory: for keeping tracking of user and char agent's conversation history
    - PromptTemplate: were used for inputting structured prompts (with retrieved context embeddings) to the LLM
- Embedded contextual data for augmented responses from LLM: Instead of using only the pre-trained knowledge of the foundational LLM, the chat agent augments the retrieved relevant knowledge chunks into the LLM prompt. Then, the LLM uses the provided contextual information to generate context-based natural language responses.
- Streamlit Front-end: We developed a Streamlit web-browser-based interface for business users to interact with the chat agent. The frontend (see Figure 1) consists of options for users configure the chat agent by using:
    - Select (dropdown boxes) and save configuration (using a button) the choices of LLM, vector store, and embedding model to be used in creating the contextual knowledge store
    - Button to upload new context document and to trigger update of the contextual knowledge store
    - A textbox (for entering custom file name) and a button to save chat session history
    - A text input box for users to type in natural language queries
    - A "Get Answer" button to trigger backend functionality to process the query and retrieve responses
    - A display area (below the thumbs up/down icons) to show the chat agent's session

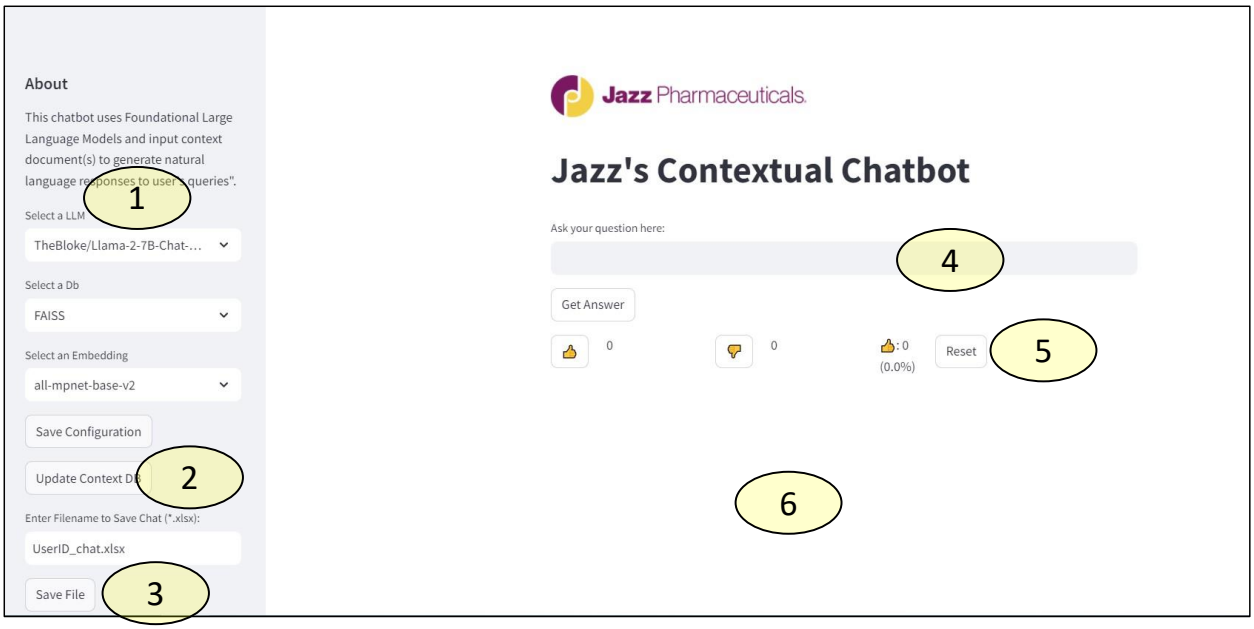


Figure 1: Frontend components of the chat agent showing 1) chat agent's configuration panel, 2) Button to update contextual knowledge base, 3) Options to save chat agent and user interaction in a file, 4) Textbox for user to input natural language queries, 5) Thumbs Up/Down, Reset icons to capture user feedback 6) Chat session window

**Results and Discussion**

We believe that assessing the reliability and response-accuracy chat agents based on natural language processing (done via the use of LLMs) has several challenges arising from retrieval-knowledge quality, contextual understanding, hallucinations, and absence of one-size-fits-all "evaluation metrics". We assessed the effectiveness of the chat agent based on its accuracy in responding to three categories of questions. There may be several factors affecting the quality of the responses generated by the chat agent using RAG approach. Some of them include,

- challenges arise from semantic mismatch: Despite using optimum embedding models, retrieved chunks of ingested documents may not perfectly align with the user query due to nuances in medical terminology, regulatory phrasing, or ambiguous language.
- Limitation of Indexing: Incomplete or outdated knowledge bases can lead to retrieval of obsolete or irrelevant documents, affecting response accuracy.
- Sparse Data in Niche/restricted/limited information domains: In specialized fields like clinical trials, limited availability of structured, high-quality, and high-volume data makes accurate retrieval difficult.
- Absence of established ground truth to compare chaat agent's responses to acceptable (human vetted) responses.

Before we present our chat agent's performance with business users, we would like to provide an important caveat. As an end user, the goal of ensuring high/acceptable accuracy from chat agent's responses requires continuous improvements in retrieval mechanisms, ingested data's context comprehension, and chat agent's response validation. Additionally, monitoring for hallucinations, refining evaluation/performance metrics, and implementing human-in-the-loop review systems are critical to enhancing chat agent's reliability. This is crucially true in high-risks/high-rewards domains like clinical trials and regulatory compliance.

**Performance Metrics**

In our research, we did not find any universally accepted, one-size-fits-all metric, vetted by subject matter experts that was cited to be useful for evaluating the performance of natural language responses generated by LLMs. As we did not have subject matter expert's recommended evaluation metrics/tool for accessing the accuracy of the responses from the chat agent, we prepared an assessment tool (document) that consisted of a set of 38 questions and their ground truth answers (written by 2 independent business users) based on the ingested document. The questions were authored by two business users and were a mix of various levels of contextual understanding of the input document. The 38 questions-set (with sensitive information replaced with "ABC") consisted of

- 14 counts of "Simple Green Category" questions made up of Fill-in-the-blank, True/False, straight forward type questions. For example,
    - Use document "ABC" for responding to this question. What is the meaning of "ABC"?
    - How often will "ABC" assessments be performed?

- 11 counts of “intermediate Orange Category” questions whose responses need, one additional level of inference before arriving at the response. For example,
    - Use document “ABC” for responding to this question. In the study design, what does the “ABC” of the study evaluate?
    - How will “ABC” status be added for all patients in both parts?
- 13 counts of “Challenging Red Category” questions requiring additional levels of logical inferences and summarizations
    - Use document “ABC” for responding to this question. What does “ABC” and “ABC” of the study evaluate?
    - Describe the “ABC” and “ABC” and “ABC” to be evaluated for “ABC” of the “ABC”?

For obtaining a numeric accuracy percentage score for the chat agent’s responses for the above outlined 38 questions, we associated a numeric value to the quality of the natural language response provided by the chat agent. User’s evaluation of the chat agent’s response was gauged using a 4-point tier-based rating system mentioned below. *Please note that the numerical value of points in our tier-based rating system do not have any significance, they are merely used to categorize types of responses generated by the chat agent.*

- 0-point category: when the chat agent provided a wrong answer
- 1-point category: when the chat agent provided a fully correct answer
- 2-point category: when the chat agent provided partially correct and partially incorrect answer
- 3-point category: when the agent provided correct but incomplete answer

We tested the agent on the assessment tool by configuring the agent to use permutations of 2 LLMs and 3 embedding models, there by obtaining 6 (LLM-Embedding) combinations of agent’s GenAI responses (results). Table-1 lists the various combinations of LLMs, embedding models and vector database used in this PoC.

| Configuration Label | Foundational LLM | Embeding Model | Vector Database |
|---|---|---|---|
| Config. A | Llama-2-7B-chat | All-MiniLM-L6-v2 | FAISS |
| Config. B | Llama-2-7B-chat | all-mpnet-base-v2 | FAISS |
| Config. C | Llama-2-7B-chat | paraphrase-MiniLM-L6-v2 | FAISS |
| | | | |
| Config. D | Llama-2-13B-chat | All-MiniLM-L6-v2 | FAISS |
| Config. E | Llama-2-13B-chat | all-mpnet-base-v2 | FAISS |
| Config. F | Llama-2-13B-chat | paraphrase-MiniLM-L6-v2 | FAISS |

Table-1: Table shows the various combinations of LLM, Embedding model and vector database used in the PoC

In our initial assessments, key metrics to assess POC’s performance were the response time and business user’s acceptability with the generated responses. Response time captures how quickly the chat agent processes queries and delivers answers, with ideal benchmarks often depending on the context. Delays in response can lead to frustration, so optimizing latency through efficient algorithms and infrastructure is critical. However, as the PoC was not built on state-of-the art (and top performing) infrastructure, we did not use response time as a performance indicator, though we acknowledge its importance as a metric for future assessment of an agent. We instead focus on business user satisfaction with PoC’s responses.

Figure 2. shows the percentage accuracies of the responses generated by the GenAI agent for each of the 6 configurations listed in Table-1.

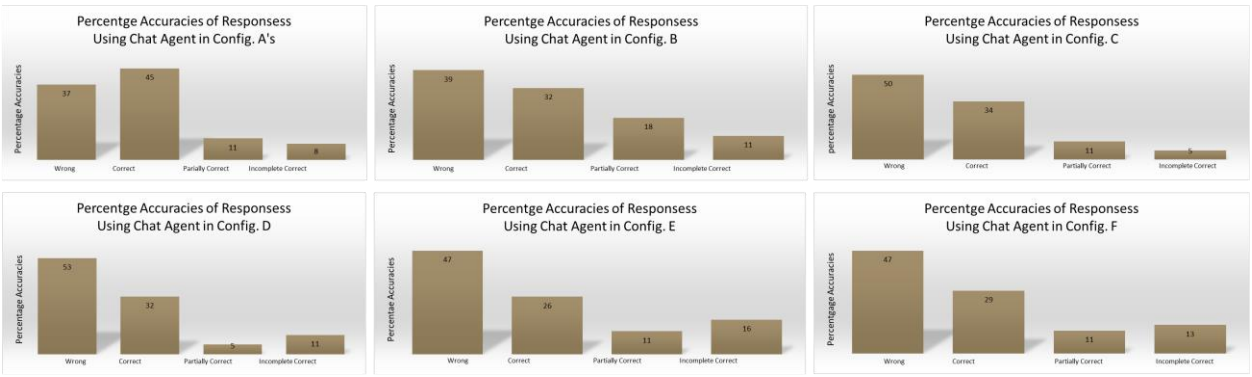


Figure 2: Histograms shows the relative percentages of accuracies for the generated responses using the chat agent in 6 confi gurations mentioned in Table 1.

Important results from Figure 2. Include

- Accuracies of GenAI tool depends on the configuration (LLM, embedding model, and vector database) used in the chat agent
- Accuracies do not always increase by switching from Llala-2-7B-chat (row-1 sub-figures in Figure 2) to Llama-2-13B-chat (row-2 sub-figures in Figure 2)
- For partially correct responses, we noticed a large fluctuation in the percentage accuracies ranging from 5% to 18%

The below Figure 3 shows a comparison of the chat agent's performance for each of the 6 configurations used in the PoC.

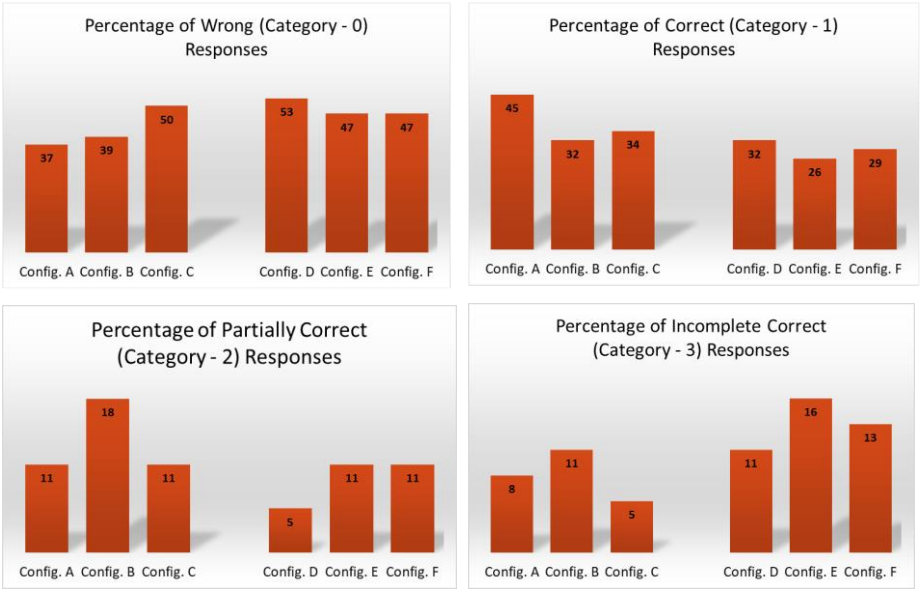


Figure 3: Comparison of 6 configurations of the chat agent for generated responses that were categorized as: wrong, correct, partially correct, incomplete correct

Important results from Figure 3. include

- Percentages of wrong, correct, partially correct and incomplete correct responses produced by the chat agent depends on the configuration (LLM, embedding model, and vector database) used in the chat agent
- For the used assessment tool, the PoC with configuration A gave the best accuracy of 45% where as configuration D got the highest score (53%) for wrong responses

**Conclusions**

In the technology space, AI is evolving very rapidly, making its way into operations of several businesses, and bringing about noticeable transitions. We believe that waiting on the sidelines and delaying (or avoiding) adoption of AL/ML based innovations in business operations may lead to lost opportunities on multiple avenues that have high return on investment. None the less, a lot of deliberate, careful risk-reward analysis, business planning and carefully drawing-up of technology-upgrade roadmaps is very important for businesses. As a starting point, we believe that businesses may engage with AI/ML on a small scale and consider inclusion of AI/ML tools as a highly recommended

agenda item in their business's evolution. It is highly advised to collaborate with AI/ML subject matter experts, business analysts, engineers, and stakeholders in businesses to identify important use cases for this newly popular technology and get a tangible hold of the return on investment brought about by inclusion of the AI/ML technology. More importantly, an assessment of the value generated post-adoption of the AI/ML tools is also very important to get an objective idea of new technology's value to the business.

A few other considerations that are important include

- Starting the AI/ML adoption at small scale on a very specific low-stake proof-of-concept (PoC) component that integrates with existing business operations
- Identifying the risks-rewards of the proposed AL/ML PoC
- Quantifying the business value of the component to the business and having a metric to assess PoC
- Considering on-premises or isolated-environment development and deployments
- Having a cautious (against) attitude on over-reliance on external APIs
- Inclusion of human-in-the loop to approve the outputs produced by the PoC
- Continuing to follow best practices that pertain to technology in the business

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Author Name: Sairam Tangirala, Ph.D., M.S. Data Science
Company: Jazz Pharmaceuticals
Email: stangirala@jazzpharma.com

Author Name: Seamus Gallivan
Company: Jazz Pharmaceuticals
Email: seamus.gallivan@jazzpharma.com

Author: Jagan Mohan Achi
Company: Jazz Pharmaceuticals
Email: jaganmohan.achi@jazzpharma.com



**References**

[1] "Transforming patient outcomes with generative AI", Renee Iacona, Francis Kendall, et.al, https://www.astrazeneca.com/what-science-can-do/topics/data-science-ai/generative-ai-drug-discovery-development.html (2024)

[2] "Intelligent Clinical Trials: Using Generative AI to Fast-Track Therapeutic Innovations", World Economic Forum, https://reports.weforum.org/docs/WEF_Intelligent_Clinical_Trials_2024.pdf (2024)

[3]" Breaking New Ground for Clinical Trials with AI/ML Applications", Richard Chau, https://www.geneonline.com/breaking-new-ground-for-clinical-trials-with-ai-ml-applications/ (2024)

[4] "Leading Artificial Intelligence (AI) Companies in Clinical Trials", https://www.clinicaltrialsarena.com/buyers-guide/artificial-intelligence-companies-clinical-trials/

[5]" Reid Hoffman Raises $24.6 Million for AI Cancer-Research Startup", Berber Jin, https://www.wsj.com/tech/ai/manas-ai-drug-discovery-reid-hoffman-93a6c023 (2025)

[6]. "A Cookbook of Self-Supervised Learning", Randall Balestriero, Mark Ibrahim, Vlad Sobal and et.al, https://arxiv.org/abs/2304.12210 (2023)

[7] "Improving language understanding by generative pre-training", Radford, A., Narasimhan, K., Salimans, T., et.al., https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (2018)

[8] "Language models are unsupervised multitask learners", Radford, A., Wu, J., et.al. , https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (2019)

[9] “Language models are few-shot learners. Advances in Neural Information Processing Systems” (NeurIPS 2020), Brown, T., Mann, B., et.al. , https://arxiv.org/abs/2005.14165 (2020)

[10] “Distilling large language models for matching patients to clinical trials”, Mauro Nievas, Aditya Basu, Yanshan Wang, Hrituraj Singh, Journal of the American Medical Informatics Association, Volume 31, Issue 9, September, pgs. 1953–1963, https://doi.org/10.1093/jamia/ocae073 (2024)

[11] “Matching patients to clinical trials with large language models”, Jin, Q., Wang, Z., Floudas, C.S. et al., Nat Commun 15, 9074, https://doi.org/10.1038/s41467-024-53081-z (2024)

[12] “Team IELAB at TREC Clinical Trial Track 2023: Enhancing Clinical Trial Retrieval with Neural Rankers and Large Language Models”, Shengyao Zhuang, Bevan Koopman, Guido Zuccon, https://doi.org/10.48550/arXiv.2401.01566 (2024)

[13] “A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions”, Lei Huang, Weijiang Yu,et.al., https://doi.org/10.48550/arXiv.2311.05232, (2023)

[14] “Hallucination is Inevitable: An Innate Limitation of Large Language Models”, Ziwei Xu, Sanjay Jain, et. Al., https://doi.org/10.48550/arXiv.2401.1187 (2024)

[15]. “The Dawn After the Dark: An Empirical Study on Factuality Hallucination in Large Language Models”, (Li et al., ACL https://aclanthology.org/2024.acl-long.586/ (2024)

[16] “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks”, Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., et.al. https://arxiv.org/abs/2005.11401 (2020)

[17]: “RAGged Edges: The Double-Edged Sword of Retrieval-Augmented Chatbots”, Philip Feldman, James R. Foulds, Shimei Pan, https://doi.org/10.48550/arXiv.2403.01193 (2024)

[18]: “FACTS About Building Retrieval Augmented Generation-based Chatbots”, Rama Akkiraju, Anbang Xu, Deepak Bora, et. Al., https://doi.org/10.48550/arXiv.2407.07858 (2024)