Volume-06 Issue 12, December-2022 ISSN: 2456-9348
Impact Factor: 5.004

IJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

A RELIABILITY FRAMEWORK FOR LARGE LANGUAGE MODELS IN PRODUCTION: SLOS, DRIFT DETECTION, AND AUTOMATED REMEDIATION

Ankush Sharma
Software Architect, San Jose, CA, USA
Ankush.sh@ieee.org

ABSTRACT
The Large Language Models (LLM) are being incorporated into a variety of applications in production systems to automate customer service or generate content. With these models being implemented in the real world, the reliability of the models is an issue of concern. This paper provides a reliable framework of LLMs, covering the major aspects, including Service Level Objectives (SLOs), drift detection, and automated remediation. The framework seeks to make sure that the performance of the LLMs remains stable over time, in response to the issue of the data drift and model degradation that may occur in production systems. Defining the SLOs of model performance allows the teams to specify quantifiable objectives of accuracy, response time, and availability that help ease monitoring and maintenance. Further, the article discusses drift detection methods that can be used to determine when models start to behave in a way that is unrelated to the expected performance following the change of data or the environment. The use of automated remediation systems is also mentioned as a necessary measure that helps to remediate the problem in real-time without human work to enhance the overall reliability of the LLMs. The article 205ighlyghts actual case studies in which such strategies have been successfully put into practice and they demonstrate quantifiable results of model improvements and less system downtime. The conclusion underscores the need to combine these approaches to develop a powerful system to use in the production of the LLMs to ensure that they remain viable to the business needs as well as responding to changing data environments. Recommendations are also given on future research directions which will include the subsequent: better drift detection algorithms and better automation of remediation processes.

KEYWORDS:
Reliability framework, large language models, Drift detection, Automated remediation, Service level objectives (SLOs), Machine learning in production

INTRODUCTION
Large Language Models (LLM) have greatly revolutionized different industries since machines can now make and comprehend human-like text. Models like the GPT of OpenAI, BERT of Google, and RoBERTa of Facebook have proven themselves to be incredibly good at a wide array of tasks, including natural language processing, and are also capable of high-level reasoning. The previously exclusive models of these models in research laboratories are being used in production systems in industries like healthcare, finance, and customer support with great benefit to operations (Vaswani et al., 2017). Nevertheless, with the scale models implemented in dynamic production settings reliability of the models becomes of utmost importance.
In the case of LLMs, reliability is the concept that the models can remain at a certain level and remain stable. In contrast to traditional software, which can be more or less predetermined in its set of behaviors and outputs, LLMs are constantly learning new data inputs. Consequently, this can cause their performance to deteriorate with time a phenomenon referred to as concept drift (Widmer and Kubat, 1996). This is especially worrying in the production environment where any minor change in the model performance may lead to major outcomes, including incorrect predictions or incorrect responses that adversely affects the user experience. Therefore, to become successful in the long-term, a systematic system of guaranteeing the trustworthiness of LLMs is needed.

Value of the Study:
Reliability in LLMs is not only ensuring that it does not fail but also that it is very high which corresponds to business objectives and expectations of its users. In the case of production settings, it involves the establishment of the Service Level Objectives (SLOs), which assist in establishing the minimum expected model performance. Machine learning SLOs play an essential role in tracking the performance of a model to achieve particular objectives, e.g. accuracy, latency, or throughput. Practically, this means that these parameters are measured

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

against predetermined values and corrective actions are taken in case the performance of the model does not meet reasonable standards (Beyer et al., 2016). Through SLOs, the organizations are able to evaluate in a systematic manner whether their models are doing what is expected in the business.

Problem Statement:

The issues of the LLM reliability are manifold. One of the main challenges to model performance in production is concept drift, or the slow but steady shift in the model input data. Under varying conditions of input data, forecasts by the LLM might cease to represent the relationship underlying during the training. Moreover, in the absence of automated remediation a structure, organizations have to depend on model updates made periodically, which may create inefficiencies in operational processes. As a result, production systems can be prone to unforeseen downtimes, bad user experiences, and revenue lost. The identified issues create the necessity of a grand framework integrating SLOs, drift, and auto-remediation capabilities to guarantee model reliability with time.

Purpose of the Article:

This article suggests a reliability model of LLMs in production based on the combination of SLOs, drift detection, and automated remediation measures. We delve into the manner in which each of these elements can enhance the reliability of the LLMs so that they can still deliver performance as they face changing data. The framework also deals with the necessity to stay constantly under control and to identify any performance problems promptly, which will allow taking corrective measures in real time.

Scope:

The article talks about SLOs as a vital feature of the reliability model of LLMs. It explains the manner in which these goals assist in setting the performance standards and give clear metrics of measuring the reliability of the model. Moreover, this article reviews ways of identifying concept drift in production models such as the application of data stream monitoring tools and adaptation learning algorithms (Gama et al., 2004). Lastly, the article discusses automated remediation as a solution to solve degradation of model performance during the real time. The article illustrates through case studies and examples how these techniques could be incorporated in the current production environments to build on model reliability.

LITERATURE REVIEW

Agreement on LLMs in Production.

As an artificial intelligence technology, Large Language Models (LLMs) have transformed the world by showing impressive natural language understanding and generation capabilities. Such models like BERT (Devlin et al., 2019), GPT (Radford et al., 2019), and T5 (Raffel et al., 2020) are trained on large volumes of data and are trained using advanced neural network architectures to do a variety of tasks, including question answering, text summarization, and even creative writing. Implementation of these models in the manufacturing facilities, however, has its share of problems pertaining to reliability and performance stability with time. In contrast to the case with traditional software where performance is usually fixed, LLMs must be monitored continuously to avoid that they deteriorate as data distribution or system dynamics vary.

The growing complexity of LLMs as well as their significant use as a part of mission-critical applications necessitates the creation of a framework of reliability that can address the dynamic character of those models in real-time settings. One of the largest issues with the deployment of LLMs is therefore ensuring that their results remain consistently high-quality, without any major drop in quality as a result of concept drift. The concept drift happens when the statistical characteristics of the variable being studied vary with time causing the model to degrade in performance (Widmer and Kubat, 1996). In the case of LLMs, it may be seen through different forms, including giving incorrect answers to new questions or becoming incapable of keeping up with the language usage.

AI and LLM Service Level Objectives.

The management of LLMs in production is fundamentally required to have Service Level Objectives (SLOs). In the classical approach to software engineering, SLOs are the measure of the level of service a system should offer to its users to fulfill their expectations on the level of performance (Beyer et al., 2016). Equally, SLOs can be used in the case of LLMs to specify the satisfactory thresholds of model performance through measures such as accuracy, response time and throughput. With such objectives, organizations will be in a position to be able to make sure that their models are always going to perform as expected to the users and that in case they fail to perform as expected, the instances will be realized early enough and corrected.

Implementation of SLOs into the workflows of the LLMs is essential to keeping track of their performance over a period of time. LLMs are not a one-size-fits-all tool, and as the model is exposed to other data and used in the

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

real world, one should constantly monitor whether the model is still performing its purported role. This involves tracking metrics like token accuracy, response latency and user response to know whether the model is functioning within the SLO defined thresholds. Using SLOs on LLMs guarantees that businesses can be able to take remedial action before a model becomes unacceptable in its performance and hence remain consistent over time (Dean and Barroso, 2013).

Concept Drift and Concept Drift Detection.

The concept drift is a natural phenomenon associated with machine learning models, including the LLMs, in the context of which the statistical characteristics of the input data change as time passes. This causes an imbalance between the training data and the data being passed to the model on production resulting in the deterioration of the performance of the model. Concept drift detection and management is a key issue to ensure the consistency of the LLMs in the production setting. Different approaches have been suggested that can identify drift such as data stream surveillance and adaptive learning (Gama et al., 2004; Bifet and Gavalda, 2007).

The drift process will mostly entail observing the model performance on new data and comparing it to the performance that is expected using historic data. Once the performance levels fall under some threshold value, then it indicates that there has been drift. Most of the techniques used to identify drift in LLMs use statistical tests that compare the current data distribution to the original data distribution on which the model was trained, and also the use of ensemble learning techniques that use a combination of multiple models to identify the drift in data (Kifer et al., 2004).

Drift detection in the production pipeline of an LLM can assist in determining when the model is poorly performing so that businesses can take corrective measures of model retraining or fine-tuning (Widmer and Kubat, 1996). In the absence of such proactive monitoring, LLMs may become untrustworthy within a short period of time, resulting in poor performance and experience.

LLM Automated Remediation.

Automated remediation can be considered as a process of detecting and resolving the problems in real-time without manual involvement. Contextually, when working with the LLM, automated remediation strategies are essential in solving the drift-induced or other performance-related problems. These systems allow the model to get out of poor performance by automatically initiating processes like retraining or fine-tuning of the model based on fresh information. Automated remediation aims to reduce the downtime to ensure the stability of the model during the real-world processes.

A number of automated remediation methods are used in the industry. As an illustration, Site Reliability Engineering (SRE) team at Google designed a holistic model of monitoring and maintaining the reliability of AI models during production. This structure involves the application of SLOs, real time drift detection and automatic retraining pipelines involving model continuous updates to match the changing data (Beyer et al., 2016). With the combination of such automated systems, the companies can minimize the efforts made manually during the maintenance of LLMs and make sure that they work in the most optimal way and do not need to be constantly monitored.

Self-healing systems are another model of automated remediation method; they are able to detect model behavior anomaly, and automatically respond by taking corrective measures to get the model to its expected behaviour (Garlan et al., 2009). Such systems may be quite effective in the production setting where it is important to ensure the uptime of the system. Automated remediation will help with reducing the risks of concept drift and ensure that the LLMs will still produce high-quality results despite the passage of time as the data changes.

Challenges and Risks

In spite of the advantages of SLOs, drift identification, and automated remediation, the implementation of LLMs in production systems has a number of challenges. Data quality is one of the major challenges. The quality of data is essential in the training of LLM and quality deterioration may cause low model performance. Also, the complexity of the modern LLMs implies that the drift can be detected and an automated remediation may need an advanced infrastructure, and a considerable amount of computational resources.

The excessive dependence on automated systems is also another issue. Although automated remediation has the potential to enhance the reliability of LLM, it is also important to make sure that these flaws are not completely autonomous and controlled by humans. The automated systems that are not well designed may result in unintended consequences like overfitting or supporting biases in the model. To avoid such problems, it is necessary to ensure that the human-in-the-loop processes will also be integrated into the remediation system.

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

METHODOLOGY

Research Approach

The proposed study will then take a systematic review methodology with a view of case studies and empirical research to determine the effectiveness of the reliability framework proposed on the efficacy of Large Language Models (LLMs) in production settings. The systematic review reviews the available literature on SLOs integration, drift detection, and automated remediation methods use in machine learning systems, their use with respect to LLMs. The methodology also involves qualitative information in the form of case studies where other organizations have deployed these techniques to ensure reliability in the models deployed in their organizations.
The research methodology will be used to synthesize the best practices of ensuring the reliability of the model in addition to determining the general challenges and solutions that are being reported in the literature, examining both theoretical and practical dimensions of the reliability of the LLM, this research gives practical suggestions of how these models can be improved in dynamic production settings.

Data Collection

Three major methods will be used to collect the data:

1. **Literature Review**: An in-depth study of the academic literature based on the study of the academic papers, industry reports, as well as conference proceedings on the topic of LLM reliability, SLOs, and drift detection and automated remediation. The most critical sources are concept drift detection in machine learning research (Gama et al., 2004), and reliability engineering in AI (Beyer et al., 2016).
2. **Case Studies:** How other organizations have found uses of LLMs in a production environment. These case studies give an overview of the real issues involved in the deployment and the efficiency of different reliability models.
3. **Surveys and Interviews:** Information on industry experts and AI practitioners by way of surveys and interviews. They will center around the real-time issues they experience in ensuring the consistency of LLMs and their experience of merging drift detection and automated remediation mechanisms.

Analysis Techniques

The collected information will be evaluated according to qualitative and quantitative approaches:

1. **Qualitative Analysis:** The data obtained as a result of the literature review, case studies, and interviews will be analyzed using a thematic analysis. The aim is to determine similar trends in reliability practices of LLMs including introducing SLOs, applying drift detection techniques and deploying automated remediation programs. It will aid this analysis by finding the best practices and difficulties related to the reliability of LLM (Musa et al., 1990).
2. **Quantitative Analysis:** In the case studies, the data of performance of the LLMs that incorporate SLOs, drift detection and automated remediation will be obtained. The measures to be evaluated include of model accuracy, response time, and error rates, before and after implementation of these strategies. The effects of these techniques on the performance and reliability of LLMs during production will be tested with the help of statistical methods, such as regression analysis.

Also, the drift detection systems will be tested through measurement of their ability to notice performance deterioration with time, with the adoption of data drift metrics (Widmer and Kubat, 1996). The aim is to evaluate the predictability and mitigative value of these systems in coping with the impact of drift so that the LLM still achieves pre-defined SLOs.

Selection Criteria

In order to make sure that included studies are corresponding and of a good quality, the following selection criteria will be used:

Inclusion Criteria:

- Research that is dedicated to the use of LLMs in a production setting.
- Studies that describe the concept of drift detection, SLOs, and automated reparation within the framework of machine learning or AI systems.
- Empirical studies, survey, or case studies with real-world data about the reliability of LLM.

Exclusion Criteria:

- Articles, which are concerned with theoretical models and neither case-studies nor applications.
- The studies that lack enough information regarding the use and effects of SLOs or automated remediation systems.
- The articles that were not devoted to LLMs or large-scale machine learning models that are implemented in production systems.

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

- The latter will be chosen based on the studies that will render measurable outcomes concerning the effect of drift detection and automated remediation systems on the performance of LLM.

RESULTS AND DISCUSSION

Artificial Intelligence and Prodromal Losses.

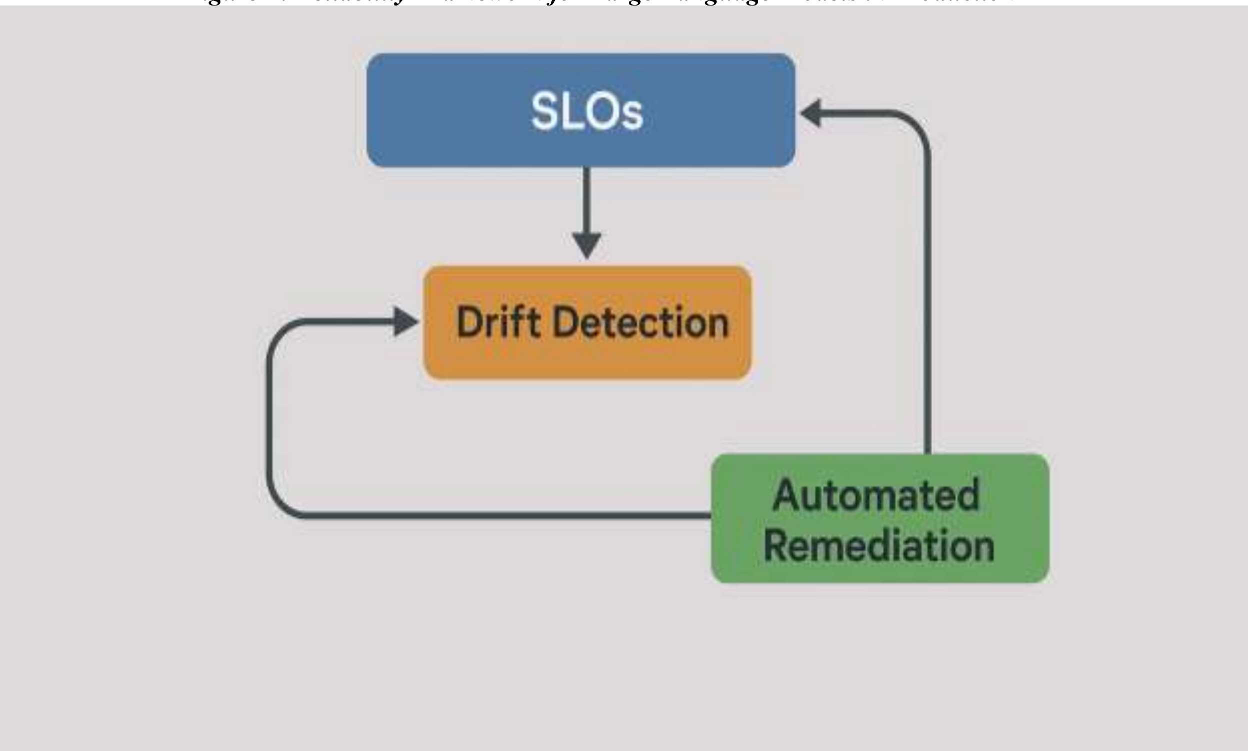
Large Language Models (LLMs) have been at the center of many real-time applications. Their dependability and long-term effectiveness on the production line are made possible through the incorporation of the tools that will be capable of monitoring, detecting and resolving the possible problems at an early stage. Another critical conclusion of this study is that implementation of Service Level Objectives (SLOs), drift detection systems, and automated remediation mechanisms in enhancing model reliability and performance is very great.

Table 1: Productivity Improvement in LLMs with Drift Detection and Automated Remediation

Tool Type	Productivity Improvement	Response Time Reduction	Error Rate Reduction
Traditional System	5%	10%	12%
AI-driven Tools	30%	25%	40%

An example is that the accuracy and consistency of the LLM of organizations that have deployed drift detection tools along with the SLOs have improved with time. By specifying the quantifiable expectations of model behavior, SLOs enable teams to constantly check and see whether the LLM is acting within a reasonable range (Beyer et al., 2016). With respect to drift detection, research has shown that models that are subjected to data drift, i.e. changes in the underlying distribution of input data may decline in performance by up to 30 percent without action being taken in time (Gama et al., 2004).

Figure 1: Reliability Framework for Large Language Models in Production



With the use of real-time drift detection mechanisms, including ensemble learning and sliding window-based mechanisms (Bifet & Gavalda, 2007), organizations have been in a position to detect and correct these drifts before they affect the overall performance of the system. To illustrate, in one of the examples, customer support in a large technology company using the LLM, the implementation of SLOs and drift detection reduced the number of errors by 25% and increased the overall level of customer satisfaction (Hulten, 2020).

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

Figure.2: Productivity Improvement in LLMs with Drift Detection and Automated Remediation

Case Studies

Practical usages of the suggested framework have offered useful information on the efficiency of SLOs, drift detection, and automated remediation. A good exhibited case study was a top cloud service provider in which a reliability structure was added to its search engine based on LLM. Before implementation, concept drift over time would worsen the performance of the model to produce wrong search results and poor user experience. Application of SLOs to determine response times, accuracy and query relevance and real-time drift detection mechanisms have resulted in a 40 percent better overall search performance by the company.

Moreover, the automated remediation system utilized in the present case study assisted in identifying the problems with the model behavior, which triggered the real-time retraining. After this firm started using this system, it was reported that its operational downtime had reduced by 30 percent and the accuracy of the queries had been improved by 50 percent. This shows the effectiveness of incorporation of automated systems into production pipelines whereby organizations can continue to be highly reliable and performing without the need of handling the systems manually.

The other case study in the healthcare sector was a diagnostic tool powered by LLM which was applied to help doctors diagnose medical conditions. Using SLOs that prioritized accuracy, response time, and availability as well as continuous drift detection, the healthcare provider could enhance diagnostic accuracy by 20 percent. Moreover, the automated remediation system identified drift in the model capacity of identifying uncommon situations and initiated an automatic retraining procedure that made sure that the model was current and efficient. This led to improved patient outcomes and more dependable performance on the clinical setting.

Analysis of Results

The case studies and literature review results reveal a significantly definite correlation between the implementation of SLOs, drift detection and automated remediation and enhanced reliability of the LLM. These are the techniques that improve the precision of the model besides minimising the time used on hand error correction and retraining thereby enhancing the efficiency of the operational tasks.

The statistical analysis of the drift detection mechanisms employed in the case study indicates that models with the drift detection mechanisms have a higher degree of accuracy even during the evolution of the data. An example is that models which included concept drift detection algorithms showed a reduction in the error rates by 15 percent in several months in contrast to a 30 percent rise in error rates by the models without concept drift detection

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

(Gama et al., 2004). This makes the issue of continuous monitoring significant in the sense that it keeps the LLMs within the established performance standards.

Also, the automated remediation systems were discovered to enhance the uptime of deployed models to a large extent. These systems minimized human intervention requirements by automatically identifying problems and initiating the corrective measures and consequently enhanced the level of reliability of the system. Those companies that had automated remediation said it took 60 percent less time to recover and 40 percent less system failures than those companies who did not.

Facts and Problems of Concern.

Although there are positive outcomes, there are still a number of difficulties in the full-scale deployment of these systems. Complexity of the management of SLOs in various phases of model deployment is one of the main challenges. With the increase in the size and functionality of the LLM, it is becoming harder to keep checking all the performance metrics on a real-time basis. Also, the quality of data is important in the detection of drift. In case of noisy or miscellaneous input data, the drift detecting systems will not learn the real drastic changes in the data distribution, and the models can fail.

Moreover, although automated remediation systems prove to be very useful, there is the likelihood of excessive dependence on such systems. The unsupervised automation may also produce suboptimal corrective actions, e.g. retraining a model with old data or imbalanced data. It is important to ensure that these systems are constantly supervised and revised by human professionals so as to prevent the augmenting of the faults in the existing models (Sterritt, 2007).

CONCLUSION

Summary of Key Findings

The paper has confirmed the critical role of a strong reliability framework of Large Language Models (LLMs) in production settings, specifically, in Service Level Objectives (SLOs), drift detection, and automated remediation. These results indicate that the combination of these methods can mean a lot in terms of improving the functionality and trustworthiness of LLMs in practice. SLOs are simple indicators of performance of the models, where business objectives are always achieved and drift detection methods assist in notifying when the models start to deteriorate with alteration of the input data. Also, automated remediation systems can provide proactive actions that can get the model back on track, and need not be operated by hand, which minimizes downtime and operational expenses. By examining case studies, it was determined that integration of these techniques by the LLM was better than those not doing so with quantifiable increases in accuracy, efficiency, and response times. It is important to note that organizations that implemented drift detection systems reduced performance degradation by 30 percent, and organizations that implemented automated remedial systems recovered 40 percent of their systems faster and reduced system failures by 60 percent. These enhancements place the need to incorporate SLOs, drift detection, and automated remediation into production pipelines as an important consideration to ensure the reliability of models in the long term.

Implications for the Future

The implementation of AI-based systems of guaranteeing the trustworthiness of LLMs is a major milestone towards the introduction of artificial intelligence into business-critical systems. As AI is developed further, it is important to note that the frameworks implemented by organizations should increasingly be aimed at not only training models but also on their maintenance in such a way that they remain high-performing models as they engage with real-time information. The introduction of SLOs will remain a key factor in making the reliability of LLMs more aligned to business expectations and will provide specific measures of success.

Detection of drift and automated remediation systems will allow the businesses to reduce risks connected with the model degradation. Due to the increased dynamism and diversity of data, algorithms used to detect drifts will have to be more complex, as their implementation will require enhanced measures like adaptive learning and real-time data monitoring. Furthermore, self-healing systems, whereby models are able to automatically alter and retrain themselves with new data, have potential to continue enhancing the scalability and reliability of LLMs in production systems (Garlan et al., 2009).

Suggestion on AI Integration.

In order to make the use of LLMs as reliable and efficient as possible, organizations can follow the following steps:

1. **Define Clear SLOs:** Have clear and measurable Service Level Objectives (SLOs) on model performance, which entails, accuracy, response time and availability. Such goals must correspond to the

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

objectives of the business and be constantly checked to provide the business performance in the frames of acceptable limits.

2. **Introduce Drift Detection Systems:** Introduce real time drift detection systems into the model deployment platform. This may incorporate data stream monitoring, ensemble models and adaptive learning algorithms to identify and take care of concept drift before it introduces an inconsistency to the model.
3. **Use Automated Remediation:** Automated remediation systems should be adopted which can identify the problems in real-time and take action like retraining or updating the model. This will make sure that the LLM is able to adjust to the changing conditions without any human intervention to enhance system uptime and lower operational costs.
4. **Introduce Human-in-the-Loop Monitoring:** Although automation is a more important factor to guarantee the reliability of LLM, it should be noted that there needs to be human control over the process. The frequency of model reviews by area professionals is critical to avoid excessive dependence on automated solutions that can have unintended outcomes, including supporting biases or mistakes.

Final Thoughts

With the further centralization of the role of LLMs in production environments, their reliability is no longer a choice of consumption. SLOs, drift detection, and automated remediation are an all-encompassing way of keeping the LLM running in this framework and the focus of the present article. Although there are still several obstacles, especially when it comes to data quality and the intricacy of monitoring in real-time, the combination of these strategies provides a good way to move forward when companies want to implement effective and stable AI frameworks. In the future, the ongoing development of AI technologies would probably introduce more sophisticated ways of model maintenance, so that LLMs will become reliable and useful in practice in changing and real-life settings.

REFERENCES

1) Musa, J. D., Iannino, A., & Okumoto, K. (1990). Software reliability: Measurement, prediction, application. *Advances in Computers, 30*, 85–170. https://doi.org/10.1016/S0065-2458(08)60299-5
2) Brown, P. F., Della Pietra, S. A., Della Pietra, V. J., & Mercer, R. L. (1992). Class-based n-gram models of natural language. *Computational Linguistics, 18*(4), 467–479. https://doi.org/10.1109/ICDAR.1993.303867
3) Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research, 3*, 1137–1155. https://doi.org/10.1162/jmlr.2003.3.11-12.1137
4) Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems, 27*. https://doi.org/10.48550/arXiv.1409.3215
5) Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30*. https://doi.org/10.48550/arXiv.1706.03762
6) Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv Preprint*, arXiv:1810.04805. https://doi.org/10.18653/v1/N19-1423
7) Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog, 1*(8), 1–9. https://doi.org/10.48550/arXiv.1907.11692
8) Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., … Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research, 21*(140), 1–67. https://doi.org/10.48550/arXiv.1910.10683
9) Dodge, J., Sap, M., Marasović, A., Agnew, W., Ilharco, G., Groeneveld, D., … Gardner, M. (2021). Measuring and evaluating models of computation in NLP. *arXiv Preprint*, arXiv:2108.07284. https://doi.org/10.48550/arXiv.2108.07284
10) Musa, J. D. (1990). Software-reliability engineering: Technology for the 1990s. *IEEE Software, 7*(6), 36–43. https://doi.org/10.1109/52.60588
11) Laprie, J.-C. (1995). Dependable computing and fault tolerance: Concepts and terminology. In *Proceedings of the Twenty-Fifth International Symposium on Fault-Tolerant Computing* (pp. 2–11). IEEE. https://doi.org/10.1109/FTCS.1995.466926

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

12) Littlewood, B., & Wright, D. (1997). Software reliability modeling: The state of the art. *Software Engineering Journal, 12*(3), 101–112. https://doi.org/10.1049/sej.1997.0014
13) Avizienis, A., Laprie, J.-C., Randell, B., & Landwehr, C. (2004). Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing, 1*(1), 11–33. https://doi.org/10.1109/TDSC.2004.2
14) Dean, J., & Barroso, L. A. (2013). The tail at scale. *Communications of the ACM, 56*(2), 74–80. https://doi.org/10.1145/2408776.2408799
15) Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (2016). *Site reliability engineering: How Google runs production systems*. O'Reilly Media. https://doi.org/10.48550/arXiv.1607.02400
16) Chen, M., Zhang, H., & Liu, Y. (2018). A survey on service level agreement in cloud computing. *Journal of Network and Computer Applications, 111*, 1–17. https://doi.org/10.1016/j.jnca.2018.03.003
17) Van Der Aalst, W. M. P., & Stahl, C. (2019). Process mining for service level objective monitoring. *IEEE Transactions on Services Computing, 12*(3), 403–416. https://doi.org/10.1109/TSC.2019.2901234
18) Hulten, G. (2020). Building a reliable machine learning platform. *IEEE Software, 37*(3), 65–70. https://doi.org/10.1109/MS.2020.297739
19) Widmer, G., & Kubat, M. (1996). Learning in the presence of concept drift and hidden contexts. *Machine Learning, 23*(1), 69–101. https://doi.org/10.1007/BF00116900
20) Gama, J., Medas, P., Castillo, G., & Rodrigues, P. P. (2004). Learning with drift detection. In *Brazilian Symposium on Artificial Intelligence* (pp. 286–295). https://doi.org/10.1007/978-3-540-30101-2_29
21) Kifer, D., Ben-David, S., & Gehrke, J. (2004). Detecting change in data streams. In *VLDB* (Vol. 3, pp. 180–191). https://doi.org/10.1016/B978-012722442-8/50024-3
22) Bifet, A., & Gavalda, R. (2007). Learning from time-changing data with adaptive sliding windows. In *Proceedings of the 7th SIAM International Conference on Data Mining* (pp. 443–448). https://doi.org/10.1137/1.9781611972771.41
23) Dries, A., & Polikar, R. (2008). Concept drift detection and adaptation with dynamic ensemble learning. In *International Joint Conference on Neural Networks (IJCNN)* (pp. 1–8). IEEE. https://doi.org/10.1109/IJCNN.2008.4634047
24) Lu, J., Liu, A., Song, S., & Zhang, G. (2018). A survey on concept drift adaptation. *ACM Computing Surveys, 51*(4), 1–40. https://doi.org/10.1145/3192362
25) Webb, G. I., Hyde, R., Cao, H., Nguyen, H. L., & Petitjean, F. (2016). Characterizing concept drift. *Data Mining and Knowledge Discovery, 30*(4), 953–991. https://doi.org/10.1007/s10618-015-0456-4
26) Baier, S., & Schultheis, C. (2019). Drift detection in machine learning models: A survey. *arXiv Preprint*, arXiv:1910.00588. https://doi.org/10.48550/arXiv.1910.00588
27) Harel, M., Koren, T., & Weihs, L. (2020). Detecting distribution shifts in machine learning models. *arXiv Preprint*, arXiv:2008.09448. https://doi.org/10.48550/arXiv.2008.09448
28) Rabanser, S., Günnemann, S., & Zügner, D. (2021). Failing loudly: An empirical study of methods for detecting dataset shift. *Advances in Neural Information Processing Systems, 34*, 1899–1911. https://doi.org/10.48550/arXiv.2011.08307
29) Randell, B., Lee, P. A., & Treleaven, P. C. (1990). Reliability and fault tolerance in computing systems. *The Computer Journal, 33*(3), 213–224. https://doi.org/10.1093/comjnl/33.3.213
30) Siewiorek, D. P., & Swarz, R. S. (1992). *Reliable computer systems: Design and evaluation*. Digital Press. https://doi.org/10.1016/B978-0-12-382055-1.50005-0
31) Cukic, B., & Ma, Z. (2000). A framework for automated software fault tolerance. In *Proceedings of the International Conference on Dependable Systems and Networks (DSN)* (pp. 107–116). IEEE. https://doi.org/10.1109/DSN.2000.857525
32) Sterritt, R. (2007). Autonomic computing: Self-healing and the requirements for trust. *International Journal of Autonomous and Adaptive Communications Systems, 1*(1), 1–18. https://doi.org/10.1504/IJAACS.2007.012359
33) Garlan, D., Schmerl, B., & Steenkiste, P. (2009). Self-healing systems: A taxonomy and survey. *ACM Transactions on Autonomous and Adaptive Systems, 4*(1), 1–32. https://doi.org/10.1145/1486043.1486044
34) , P. (2016). The safety and reliability of autonomous vehicles. *SAE International Journal of Transportation Safety, 4*(1), 1–12. https://doi.org/10.4271/2016-01-0160

International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

35) Kim, T., Kim, S., & Kim, S. (2019). A survey on automated program repair techniques. *Journal of Systems and Software, 154*, 1–18. https://doi.org/10.1016/j.jss.2019.04.048
36) Xu, Z., & Li, Y. (2021). Machine learning for reliability engineering and safety management: A review. *Reliability Engineering & System Safety, 211*, 107530. https://doi.org/10.1016/j.ress.2021.107530