YOUR FIELD · SUPPORTING DOMAIN

Administrative, managerial and collaboration activities in Engineering – Personal administrative efficiency of Engineers

This section explores how generative AI enhances administrative and collaborative tasks in engineering, focusing on AI assistants, technical writing automation, code documentation, and report generation. Key advancements include the automation of technical queries, documentation generation, and improved code commenting, all aimed at increasing productivity and collaboration in engineering teams.

Christophe Pennetier, André-Benoît De Jaegere

Generative Engineering · Volume I · written November 2024 to February 2025, two years after ChatGPT; its text is kept as published · https://generative-engineering.aquia.studio/fields/administrative/

CONTENTS

Overview 1
AI Assistants for Engineering Tasks 2
How are AI assistants specifically helping with technical queries in engineering 2
What advancements have been made in automating technical writing tasks for engineers? 3
Documentation and Report Generation 5
How does generative AI improve code documentation and commenting? 5
Citations for Section on Administrative and Managerial Collaboration 6
General References for Section on Administrative and Managerial Collaboration 7

Overview

Generative AI is transforming the field of engineering through its applications in natural language processing (NLP) across several key categories. These include AI assistants for engineering tasks, documentation and report generation, and code documentation and commenting. Here's an overview of how generative AI is being utilized in these areas:

AI Assistants for Engineering Tasks

Generative AI-powered virtual assistants are increasingly being integrated into engineering workflows to handle technical queries and automate routine tasks. These AI assistants leverage large language models (LLMs) to provide real-time support, enhancing productivity and efficiency. For example, companies like Siemens have developed industrial copilots that integrate AI into CAD software to automate routine tasks, provide suggestions, and detect potential errors early in the design process[1]. This integration not only boosts productivity but also reduces costly design mistakes.

How are AI assistants specifically helping with technical queries in engineering

Focusing on **technical queries**, AI assistants leverage LLMs to offer instant and precise help—ultimately freeing engineers to tackle higher-level, creative challenges.

AI assistants are significantly enhancing the way engineers handle technical queries by leveraging generative AI and large language models (LLMs). Here are some specific ways these AI assistants are helping:

Instant Technical Support

AI assistants like Kapa.ai are designed to provide reliable answers to complex technical questions posed by developers and software end users. These systems utilize LLMs to process and understand technical documentation, allowing them to offer accurate and contextually relevant responses. This capability is particularly beneficial in environments where quick access to precise information is crucial, such as in software development and IT support[2].

Integration with Engineering Tools

AI assistants are being integrated into engineering environments to streamline workflows. For instance, Beckhoff's TwinCAT Chat is embedded within the TwinCAT XAE engineering environment, enabling engineers to receive instant answers, generate code, and configure hardware setups directly within their development tools. This integration not only enhances

productivity but also allows engineers to focus on more creative and complex tasks by automating routine queries and configurations[1].

Enhanced Documentation Searches

AI-driven tools simplify access to engineering documentation by performing advanced searches that increase the pace of learning and information retrieval. This capability helps engineers quickly find the information they need without manually sifting through extensive documentation, thus improving efficiency and reducing the time spent on research[1].

Customization and Accuracy

AI assistants like Kapa.ai emphasize accuracy in their responses by using retrieval augmented generation (RAG) techniques, which enhance LLM performance by integrating relevant external data sources. This approach ensures that the answers provided are accurate and based on a comprehensive understanding of the available documentation. Such precision is critical in maintaining trust with users, especially when these systems are deployed externally to end users[2].

Overall, AI assistants are transforming how engineers manage technical queries by providing instant support, integrating seamlessly with existing tools, enhancing documentation searches, and ensuring high accuracy in responses. These advancements not only improve productivity but also enable engineers to focus on more innovative aspects of their work.

What advancements have been made in automating technical writing tasks for engineers?

Shifting to **technical writing**, Generative AI tackles documentation bottlenecks—streamlining content creation, ensuring consistency, and supporting iterative updates in real-time.

Advancements in automating technical writing tasks for engineers have been propelled by the integration of generative AI technologies, which streamline the creation, maintenance, and distribution of technical documentation. Here are some key advancements:

Automation of Routine Tasks

Generative AI tools like ChatGPT and Mintlify automate repetitive and time-consuming tasks in technical writing. These tools can generate documentation from code, update content in real-time as changes occur, and ensure consistency across multiple documents. This automation allows technical writers to focus on more strategic aspects, such as enhancing user engagement and making technical content more accessible to non-technical users[3][6].

Improvement in Content Quality

AI-powered tools enhance the quality of technical documentation by identifying gaps, suggesting improvements, and ensuring consistency in tone, style, and terminology. For example, tools like Acrolinx provide proactive edits to improve clarity and correctness. This results in more accurate and valuable documentation that meets industry standards[5][6].

Personalization and User-Centric Documentation

Generative AI enables the creation of personalized documentation tailored to specific user needs. By analyzing user behavior and preferences, AI can generate customized tutorials, FAQs, and troubleshooting guides. This approach not only enhances user satisfaction but also reduces the learning curve associated with new software products[6][7].

Integration with Existing Tools

AI-driven documentation tools are designed to integrate seamlessly with popular development environments and platforms like GitHub. This integration allows for the automatic generation or updating of documentation directly within these environments, improving workflow efficiency[3][8].

Enhanced Collaboration and Accessibility

Automated documentation systems facilitate better collaboration among teams by providing a centralized repository for all documents. This ensures that team members can access and update documents in real-time, reducing bottlenecks and fostering a more cohesive workflow[4][5].

Predictive Analytics and Continuous Improvement

AI's predictive analytics capabilities allow for the identification of trends in how users interact with documentation. By analyzing data on frequently

accessed sections or areas where users encounter difficulties, AI can inform continuous improvement strategies, helping prioritize updates that provide significant benefits to users[6].

Documentation and Report Generation

Taking a **broader perspective** on documentation, generative AI also tackles report generation—consolidating data and findings into coherent narratives that expedite decision-making.

Generative AI is also making significant strides in automating technical writing tasks. LLMs can automate the creation of model documentation by synthesizing information from diverse sources into coherent narratives. This approach accelerates production timelines and enhances document quality by reducing the manual effort required from subject matter experts (SMEs)[9]. For instance, EY has explored using generative AI for model documentation automation, which involves extracting information, analyzing data, and generating narratives that are then reviewed by SMEs for accuracy and relevance[9].

How does generative AI improve code documentation and commenting?

Finally, in **software-centric engineering tasks**, Generative AI streamlines code documentation, bridging technical gaps and fostering better collaboration among development teams.

Generative AI significantly enhances code documentation and commenting by automating the creation of clear, comprehensive, and up-to-date documentation. Here are some key advancements:

Automated Documentation Generation

- **Natural Language Processing (NLP):** Generative AI uses NLP to generate human-readable documentation directly from code. This includes creating docstrings, comments, and explanations that are easy for developers to understand, making technical knowledge more accessible within teams[10][11].

- **Real-Time Updates:** AI tools can automatically update documentation as code changes, ensuring that documentation remains current and reducing the manual burden on developers[10][12].

Improved Code Comprehension

- **Explanatory Comments:** AI can generate detailed comments that clarify complex functions and algorithms, aiding in code readability and maintenance. This helps developers quickly grasp the purpose and functionality of code segments[11][13].
- **Contextual Understanding:** By analyzing existing codebases, AI can provide context-aware suggestions that align with coding standards and best practices, enhancing consistency across projects[11][13].

Efficiency and Consistency

- **Time Savings:** Automating the documentation process reduces the time developers spend on writing and updating documentation, allowing them to focus on more critical tasks[3][12].
- **Error Reduction:** AI-generated documentation minimizes human error, ensuring accuracy and completeness in documenting code functionalities[11][13].

Integration with Development Tools

Seamless Integration: Many AI tools integrate with popular development environments like GitHub, allowing for the automatic generation of documentation directly within these platforms. This integration supports continuous documentation updates in sync with code changes[8][13].

Overall, generative AI transforms code documentation by making it more efficient, accurate, and accessible. These advancements not only improve developer productivity but also enhance the overall quality of software projects.

Citations for Section on Administrative and Managerial Collaboration

[1] Design Engineering - Industrial Gen AI and Copilots Manufacturing
[2] TechCrunch - How Kapa Uses LLMs to Help Companies Answer Users' Technical Questions Reliably
[3] PaceAI - Generative AI for Technical Documentation

[4] Secoda - The Need for Automated Documentation
[5] Fluid Topics - Will AI Replace Technical Writers?
[6] Taliun - Revolutionizing Technical Writing in Software Development with AI
[7] LinkedIn - Future Technical Writing Trends: Predictions by Samir Senapati
[8] CodeMotion - 10 Documentation Tools You Must Try in 2024
[9] EY - Documentation Automation Using Generative AI
[10] LeadDev - Will ChatGPT and Generative AI Replace Internal Code Documentation
[11] Tabnine Blog - AI Code Documentation: Why Documentation Hurts and How AI Helps
[12] DevOps.com - How Generative AI Can Streamline Code Documentation
[13] ScreamingBox - 6 Essential AI Tools for Code Documentation

General References for Section on Administrative and Managerial Collaboration

[G1] arXiv - Paper 2310.18648
[G2] Waterstechnology - SP Debuts Spark Assist GenAI Copilot
[G3] CNBC - JPMorgan Chase AI Assistant
[G4] LinkedIn - Probable Impacts of Automation on Technical Writing
[G5] EY - Documentation Automation Using Generative AI