ADMI IJST International Journal Science and Technology

OPEN ACCESS

IJST Vol 4 No. 2 | July 2025 | ISSN: 2828-7223 (print), ISSN: 2828-7045 (online), Page 67-75

Analysis Large Language Model (LLM) for Digital Transformation in a Ministry

Andre Pratama Adiwijayaa
Faculty of Economics, Gunadarma University, Indonesia

Article History
Received : 24 June 2025
Revised : 17 July 2025
Accepted : 21 July 2025
Published : 23 July 2025

Corresponding author*:
andre64@staff.gunadarma.ac.id,

Cite This Article:
Andre Pratama Adiwijaya. (2025). Analysis Large Language Model (LLM) for Digital Transformation in a Ministry. *International Journal Science and Technology*, *4*(2), 67–75.

DOI:
https://doi.org/10.56127/ijst.v4i2.2194

Abstract: Digital transformation in government requires the use of cutting-edge technology to improve efficiency and accuracy in decision-making. Ministries are strategic institutions that manage large-scale data and information, particularly related to regional government administration and population affairs. This article examines the application of Large Language Models (LLM) as an artificial intelligence (AI)-based solution to assist document processing, information extraction, and policy analysis. The study presents a relevant legal framework, LLM model recommendations, a schematic diagram of LLM implementation from end-to-end, and Python code examples for technical simulations. This approach is expected to strengthen the Ministry's digital system in accordance with the guidelines of the Electronic-Based Government System (SPBE) and One Data Indonesia.

Keywords: LLM, AI, Python, Ministry

INTRODUCTION

The Ministry plays a vital role in regulating government affairs in Indonesia, including population administration, regional government development, regional financial oversight, and regional planning. As a strategic institution, the Ministry manages large amounts of complex data spread across various levels of government.

However, to date, data digitization within the Ministry still faces significant challenges. One major issue is the continued dominance of physical documents, whether in the form of manual reports, correspondence, or local government document archives. Many regional agencies have not yet fully transitioned to integrated electronic systems, resulting in manual data collection from the regions to the central government, which is often slow and potentially leads to duplication or loss of important information.

Furthermore, there is fragmentation in the applications and information systems used within ministries and local governments. For example, there are numerous separate systems for population data management, regional financial systems, regional budget reports, and regional information. Often, these systems are built by third parties with varying data standards and are not interconnected. As a result, data integration across applications is difficult, slowing down analysis and decision-making at the ministerial level.

This condition shows the need for technological solutions that are capable of:

- Intelligently automate the extraction of information from physical and digital documents,
- Efficiently combine and interpret data from multiple sources, and
- Presenting structured and easy-to-understand information for policy makers.

One promising approach is the application of *Large Language Model (LLM),*namely an artificial intelligence model based on natural language processing (NLP). LLM has the ability to read, understand, and summarize documents in text format, including PDF documents, policy reports, and other unstructured data. By using *LLM,* The ministry can automate the process of identifying important information, integrating data from various sources, and providing a data-based question-and-answer and reporting system.*Ai*r esponsive.

Implementation *LLM* not only accelerates digitalization, but also strengthens bureaucratic efficiency in realizing the Electronic-Based Government System (SPBE) and supports the national One Data Indonesia program. Therefore, this study focuses on how the model *LLM* can be implemented strategically in ministries, accompanied by a relevant technical implementation framework and legal basis.

RESEARCH METHOD

This research employs a descriptive-qualitative approach, incorporating literature review, a study of existing information systems within the Ministry, and technical simulations using Python programming and Large Language Models (LLM). The research also includes system integration engineering, specifically how multiple applications or data from different units can be consolidated into an AI-based pipeline.

2.1 Research Stages

The following are the stages used in this research:

No	Research Stage	Activities
1	Literature and Regulation Study	Reviewing the Law, Presidential Decree, and Ministerial Regulation regarding SPBE and One Data
2	System & Problem Identification	Analyzing existing applications at the Ministry of Home Affairs and their integration constraints
4	Data Integration Design	Designing application and data integration architecture schemes
5	LLM Model Simulation	Experiments with GPT-4 and LLaMA models for NLP tasks Experiments with GPT-4 and LLaMA models for NLP tasks
6	Analysis & Evaluation	Assessing the effectiveness of the LLM model in reading, extracting, and compiling reports

This research maximizes the function of LLM or machine learning to be able to read and understand all ministerial documents or applications within the ministry in order to achieve maximum integration and become a reference for decision makers to make comprehensive decisions.

RESULT AND DISCUSSION

Application Integration Scheme & LLM Creation Flow

In this research, the initial process for creating an LLM is to integrate applications from application 1 to documents in PDF or text format. The integration method used in this research is the format*Data Integration Middleware*which acts as a bridge between all applications and documents that will be included in Bigdata. Furthermore, in*Data Integration Middleware* There is a data standardization process and data recognition for the machine. The method used in this research is ETL to provide context for each piece of data that will later be processed by the system or LLM. The detailed diagram is in Figure 1.

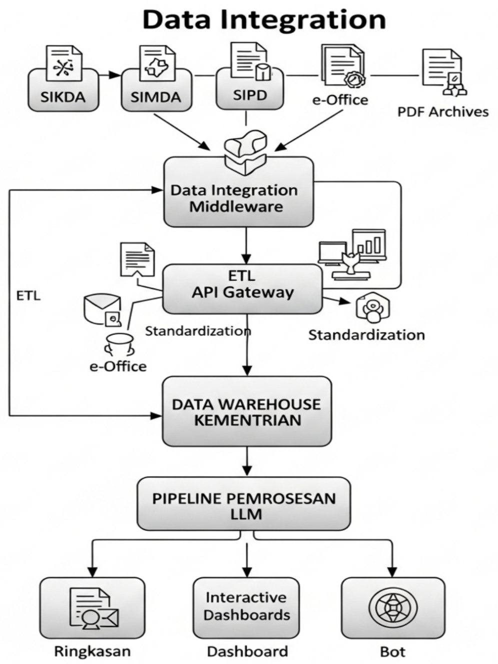

Figure 1. LLM Creation Diagram

The next process is the creation of a model from the data that has been entered into the data warehouse. After several sets of models have been created, the results of the LLM will be in the form of a document summary.*dashboard interactive*and BOT (Chat).

LLM Model Development Flow

The development of the LLM model for the internal needs of the Ministry of Home Affairs began with the collection of domain-specific data, namely government documents commonly used within the Ministry of Home Affairs and local governments. This data includes APBD accountability reports, Musrenbang results documents, meeting minutes, regional regulations, assignment letters, regional head performance evaluation documents, and official official documents in digital form (PDF, Word), or scanned forms that require OCR (Optical Character Recognition). The main objective of this data collection is to build a training corpus that reflects the characteristics of Indonesian bureaucratic language and the government context.

The next stage is data cleaning and preprocessing. This process involves converting non-text documents into clean text, removing unnecessary characters, reordering paragraphs damaged by OCR, and normalizing the format and structuring the dataset. This is followed by tokenization, which involves breaking the text into word units (tokens) and removing stop words such as "and," "or," and "yang," which are deemed meaningless for modeling. The final corpus is then stored in JSON or JSONL format, which can be used for model training or fine-tuning.

In the model selection and training phase, there are two main approaches: (1) using an external API-based LLM model such as OpenAI's GPT-4, which can be used directly without a local training process, or (2) using an open-source model such as LLaMA 2/3, BLOOM, or IndoBERT, which allows for local fine-tuning on ministry servers. For specific government needs, models such as IndoBERT and LLaMA 3 are considered suitable because they can be customized with internal data and run on-premises to ensure data confidentiality. Fine-tuning is carried out using supervised learning using a prompt-completion pair format (for example: the prompt is the instruction "Summarize the contents of the following APBD report," and the completion is an ideal summary of the report).

Once the training or model development process is complete, the LLM can be deployed as an internal API service, a chatbot, or a module within an interactive dashboard. At this stage, the model is tested to perform tasks such as entity analysis (NER), document type classification, summarization, and automated responses to policy questions (Q&A). The model's output is presented to users in the form of an interactive chatbot, automated reports, or integration into a dashboard system like Power BI. This process demonstrates that the LLM is not just a plain text tool, but also an administrative knowledge transformer that can bridge the gap between big data and rapid AI-based decision-making in the public sector.

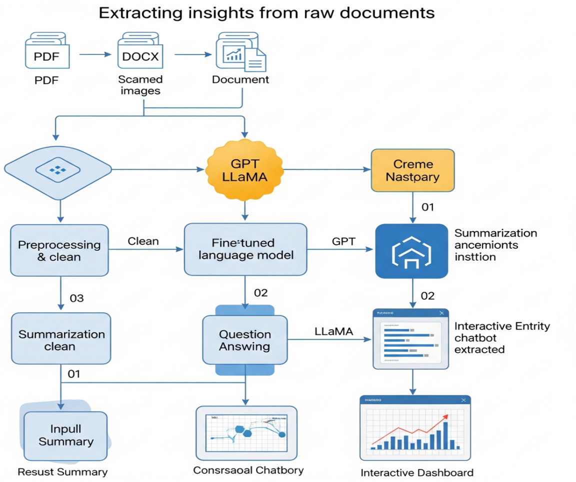

Figure 2. LLM Development Flow

In the LLM development flow process depicted in Figure 2, which emphasizes the process*fine-tuning*The model then performs summaziation to obtain results according to the questions that will be asked to the bot.

Simulation Results of Phyon Engineering and Open AI API

A technical simulation was conducted to evaluate the ability of a Large Language Model (LLM) to read and analyze government documents using an API-based approach. This study used OpenAI's GPT-4 as the primary model, accessed through the OpenAI Chat Completion API. The programming language used was Python, with the OpenAI library, PyMuPDF for PDF document reading, and Langchain for prompt management. The test documents included regional budget accountability reports, circulars, and regional regulation (Perda) documents in PDF format.

The first step involves reading a PDF document using PyMuPDF, then chunking the document into text chunks of 1500–2000 tokens to maintain the model's context. These chunks are then iteratively sent to the GPT-4 API via the Chat Completion function. Each prompt is given explicit instructions such as: "Summarize the contents of the following report in 3 paragraphs" or "What is the essence of this regulation?" to test summarization and entity extraction capabilities. The resulting summaries are then saved in structured text format for further analysis.

An example of a simulation result on a document titled "Report on the Realization of the Regional Budget for Regency X in 2023" shows that GPT-4 is able to extract important information such as the amount of regional spending, priority programs, and obstacles to program implementation. The model is also able to recognize the document's structure, even though the PDF format is not always standardized. In another simulation using a circular letter, the model successfully identified policy orders, the target agency, and the effective date of the regulation. The model demonstrated high semantic accuracy, especially in answering specific questions from long documents parsed into text chunks.

The Q&A (Question Answering) feature was also tested to evaluate how well the model understood the local government context. Questions such as "When is the budget reporting deadline?" or "What are the priority programs in the health sector?" were asked based on test documents. The model provided relevant and factual responses, with an average response latency of <10 seconds per request. The GPT-4 model was also able to understand government terms such as "capital expenditure allocation," "budget refocusing," and "regional fiscal support," demonstrating a semantic match to the Ministry of Home Affairs context.

Although the results are promising, there are limitations such as overly general responses when the data in the document is less explicit. Therefore, in the production phase, it will be necessary to adjust the engineering prompt and integrate with external metadata (e.g., document ID, sending agency, or regulatory type). This simulation demonstrates the potential of using LLMs such as GPT-4 to digitize government document processing, especially in the context of summarization, classification, and text-based question and answer services. A total of 79 virtual machines were used, divided as follows:*Database, Kube – Pool dan Management & Network* Figure 4 explains the details of using all virtual machines.

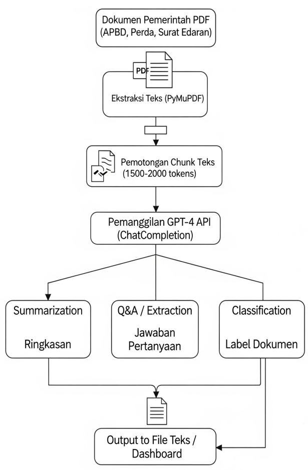

Figure 4 Python Engineering Simulation Flow + OpenAI API

CONCLUSION

Digital transformation within ministries requires a technology strategy that adapts to bureaucratic complexity and the large volume of administrative documents. This study demonstrates that the application of Large Language Model (LLM) technology can provide a concrete solution to address the challenges of manual document processing, fragmented data integration, and the difficulty of presenting information quickly and accurately. LLMs such as GPT-4, LLaMA 3, and IndoBERT have been proven capable of performing natural language processing (NLP) tasks such as data extraction, summarization, document classification, and question-and-answer services, which are highly relevant to supporting public services and policy-making across various ministries.

Technical simulations using Python and the OpenAI API also demonstrated that LLM can be integrated into the ministry's internal systems with high efficiency. The model can read financial reports, extract critical information, and answer policy questions from digital documents alone. This capability offers significant advantages in accelerating policy analysis without requiring manual reading of entire documents. With the right infrastructure, LLM can also be used to build monitoring dashboards, service chatbots, or automated reporting systems for work units at the central and regional levels.

However, for optimal implementation of LLM, the ministry needs to prepare a supportive digital framework. This includes integrating information systems across units into a single national data platform, digitizing physical archives, ensuring regulatory-based data security (e.g., classifying state confidential documents), and training human resources in AI and data literacy. Furthermore, open-source models such as IndoBERT and LLaMA need to be fine-tuned to be more sensitive to local contexts and domestic policies. Model selection should also consider storage location (on-premises/cloud), GPU availability, and long-term maintenance support.

As a suggestion, ministries can initiate LLM implementation through pilot projects in several strategic directorates with high document burdens. LLM use can be focused on summarizing policy reports, detecting late reporting, and providing chatbot-based public services. Development can be carried out in stages while establishing data security standards and interoperability between applications. If managed well, LLM integration within ministries will be a significant leap toward smarter, faster, and more scalable governance in support of the Electronic-Based Government System (SPBE).

REFERENCES

BigScience Workshop. (2022). *BLOOM: A 176B-parameter open-access multilingual language model*. Hugging Face. https://huggingface.co/bigscience/bloom

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems, 33*, 1877–1901. https://doi.org/10.48550/arXiv.2005.14165

Cahyawijaya, S., Winata, G. I., Wilie, B., Vincentio, K. A., Li, Z., Fung, P., & Purwarianti, A. (2021). IndoBERT: A pretrained language model for Indonesian. In *Findings of the Association for Computational Linguistics: EMNLP 2021* (pp. 1032–1038). https://aclanthology.org/2021.findings-emnlp.89/

Chollet, F. (2021). *Deep learning with Python* (2nd ed.). Manning Publications.

Hugging Face. (2024). *Transformers documentation*. https://huggingface.co/docs/transformers/index

Indonesia Digital Government. (2023). *Guidelines for SPBE implementation in ministries/institutions and regional governments*. https://spbe.go.id

Law No. 11 of 2008 concerning Electronic Information and Transactions (ITE). (2008). Republic of Indonesia.

Law No. 23 of 2014 concerning Regional Government. (2014). Republic of Indonesia.

Minister of Home Affairs Regulation No. 70 of 2019 concerning the Regional Government Information System (SIPD). (2019). Ministry of Home Affairs of the Republic of Indonesia.

OpenAI. (2023). *GPT-4 technical report*. https://openai.com/research/gpt-4

Presidential Regulation of the Republic of Indonesia Number 39 of 2019 concerning One Data Indonesia. (2019). https://satudata.go.id

Presidential Regulation of the Republic of Indonesia Number 95 of 2018 concerning the Electronic-Based Government System (SPBE). (2018). https://jdih.setneg.go.id

Raschka, S., & Mirjalili, V. (2022). *Machine learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python*. Packt Publishing.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A., Lacroix, T., ... & Scao, T. L. (2023). LLaMA: Open and efficient foundation language models. *Meta AI Research*. https://arxiv.org/abs/2302.13971