iNeuron

Architecture Design

Document Tagging Project

Author Name: Dibyendu Biswas.

Revision Number: 1.0
Last Date of revision: 27-September-2023

Contents

Abstract

The Document Tagging Project is an innovative and scalable solution designed to address the challenge of organizing and categorizing vast amounts of unstructured textual data. In today's data-driven world, organizations accumulate massive repositories of documents, articles, reports, and other textual content. Efficiently tagging and classifying these documents based on their content is essential for improving searchability, knowledge management, and information retrieval.

The impact of the Document Tagging Project extends across various industries and applications. It empowers content creators, knowledge managers, and data analysts to streamline document organization and retrieval processes, thereby saving time and improving decision-making. Additionally, the project's scalability ensures adaptability to diverse data sources and domains, making it a valuable asset for organizations seeking to unlock the full potential of their textual data.

1 Introduction

1.1 Why this Architecture Design Document

Architecture design, in the context of software development and system engineering, refers to the process of defining the overall structure, components, modules, and relationships of a system or application. It is a critical phase in the software development life cycle and lays the foundation for building complex, scalable, and maintainable software solutions. The primary goal of architecture design is to create a blueprint that guides developers in constructing a system that meets both functional and non-functional requirements.

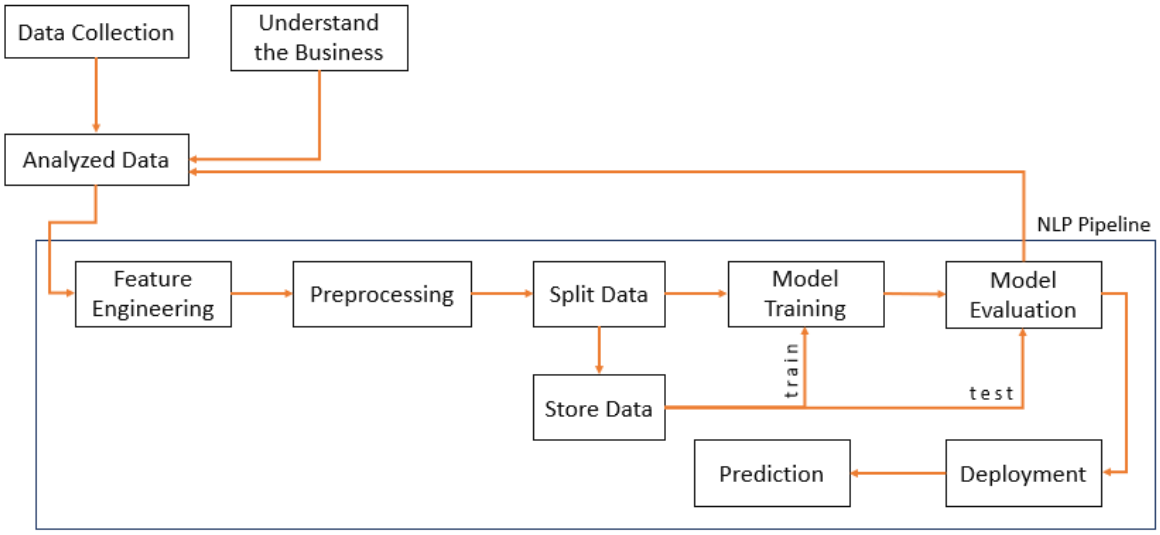
The Architecture Design will do:

Architecture design serves several crucial purposes in the development of software systems and complex applications:

- **Structural Blueprint:** It provides a high-level structural blueprint of the system, defining its components, modules, and how they interact. This helps in visualizing the system's organization.
- **Guidance for Development:** Architecture design guides developers by specifying the framework and guidelines within which they should work. It sets the boundaries and rules for building the system.
- **Alignment with Requirements:** It ensures that the system's architecture aligns with the functional and non-functional requirements, meeting the needs of stakeholders and users.
- **Scalability and Performance:** Architecture design considers scalability and performance requirements, allowing the system to grow to accommodate increased workloads while maintaining optimal performance.
- **Reliability and Availability:** It incorporates mechanisms for ensuring the system's reliability and availability, minimizing downtime and disruptions.
- **Security and Privacy:** Security measures are integrated into the design to protect against vulnerabilities and unauthorized access, safeguarding sensitive data.
- **Maintenance and Extensibility:** The design emphasizes modularity and maintainability, making it easier to update and extend the system as requirements evolve.
- **Cost-Efficiency:** It helps in optimizing resource utilization and minimizing infrastructure costs, ensuring efficient use of resources.
- **Risk Mitigation:** Architecture design allows early identification and mitigation of risks associated with the development process, reducing the likelihood of major issues later.
- **Basis for Evaluation:** The design can be used as a basis for evaluating the feasibility, cost, and technical viability of the project.
- **Quality Assurance:** It supports quality assurance efforts by defining design patterns, coding standards, and best practices of ensure the final product meets quality expectations.

2 Design Details

2.1 Process Flow



2.2 Event log

The system should log every event so that the user will know what process is running internally.

Initial Step-by-Step Description:

- The system identifies at what step logging required.
- The system should be able to log each and every system flow.
- Developer can choose logging method (training_logs and prediction_logs).
    - training_logs is logging the training pipeline.
    - prediction_logs is logging the prediction pipeline.

- System should not hang even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

2.3 Error Handling

Should errors be encountered, an explanation will be displayed as to what went wrong? An error will be defined as anything that falls outside the normal and intended usages.

2.4 Performance

Document-Tagging project is to develop a robust and efficient system for automatically assigning relevant tags or labels to documents based on their content. It categorizing and organizing a large corpus of documents, making it easier for users to search, retrieve, and manage information. Also, model retraining is very important to improve the performance.

2.5 Reusability

The code written and the components used should have the ability to reused with no problems.

2.6 Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

2.7 Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

2.8 Technical Stack

2.9 Deployment

Either I can deploy on AWS or Heroku or both.

3 Conclusion

Document Tagging project represents a significant step forward in automating and optimizing the categorization and organization of documents within organizations. By leveraging machine learning and natural language processing techniques, this project offers a powerful solution for improving document management workflows.

Through the development of a robust and scalable system, users can efficiently assign relevant tags to documents, enabling easier search, retrieval, and organization. The multi-label classification capability ensures that documents can be associated with multiple categories or topics, reflecting the complex nature of real-world document collections.

The project's user-friendly interface and customization options empower users to tailor the tagging system to their specific needs, whether it's in a corporate setting, academic institution, or any knowledge-intensive organization. Integrating the system with existing document management tools further enhances its utility and impact.

With a focus on accuracy, performance optimization, and ongoing maintenance, the Document Tagging project aims to provide a reliable and efficient solution that streamlines document management processes, saves time, and boosts productivity.

---------- **Thank You**