Technical Report: Assignment 02

DS 5110: Introduction to Data Management and Processing
Fall 2024

Contents

1 Project Kickoff

1.1 What are the specific goals of this project?

The main goals are to ensure reliable data quality monitoring and implement anomaly detection for IoT sensors in a smart home environment. This would involve:
i. Detecting faulty or missing sensor data.
ii.Identifying abnormal sensor behavior.
iii. Enhancing data consistency for downstream analysis and decision-making.

1.2 How do we define the project scope clearly to avoid scope creep?

i. Identify Core Deliverables: Focus on specific sensor types and smart home systems (e.g., temperature, motion, security sensors).
ii. Clear Boundaries: Limit the number of anomalies to detect (e.g., missing data, extreme values).
iii. Stakeholder Alignment: Make sure the team agrees on the project's boundaries.
iv. Regular Reviews: Hold checkpoints at each milestone to reassess the scope and ensure no new features are added without consideration.

1.3 What deliverables must be completed at different phases?

Phase 1: Project Plan, Requirements Gathering, and Initial Dataset Identification.
Phase 2: Data Preprocessing
Phase 3: Anomaly Detection (Rule-based, Machine Learning).
Phase 4: Testing, Validation
Phase 5: Visualization Tools
Phase 6: Final Report

1.4 What are the major milestones, and what deadlines should we set?

Project Plan Scope Agreement (October 18).
Data Collection Preliminary Analysis (October 25).
Model Selection Implementation (November 15).
Testing and Validation (November 22).
Final Deliverables and Presentation (Week Nov 30).

1.5 Do the team's capabilities align with these goals? Are there any gaps that need to be addressed early on?

Problem: In terms of preprocessing and machine learning, the team has adequate experience and knowledge. Domain knowledge is the only area where the team would have to gain understanding into IoT sensors' function and expected data patterns.

1.6 Do you have a dataset ready to use for the current project?

The team has identified a datset from ARAS Datasets, collected by boğaziçi university in turkiye. It consists of sensor data from a smart home system in multiple houses.

2 Team Discussions

2.1 What are the core skills each team member brings to the table?

All the members of the team are adept at using python. Jay has strong experience in building interactive dashboards Asmita has prior experience at anomaly detection, and model implementation. Jyothssena has experience with sensor data collection, consistency and preprocessing.

2.2 How will each person's expertise contribute to specific tasks?

Jyothssena's experience with sensor data will help the preprocessing stage run smoothly. Asmita's skills would help experiment with different techniques for anomaly detection. Jay's expertise at visualization would speed up reporting.

2.3 What skills are missing that may cause delays or challenges?

The team has not worked with anomaly detection , and the algorithms associated with that are new to the team.

2.4 What tools do we have experience with, and what do we need to learn?

The team is very experienced in python and sql. However Cloud platforms may be one region to gain more knowledge.

2.5 What programming languages and platforms should we select based on our project needs and team experience?

i. Python/R for data analysis and ML. ii. Cloud platforms like AWS, Azure for deploying IoT solutions. iii. Databases like SQL/NoSQL.

3 Skills Tools Assessment

3.1 Are there external resources or team members with expertise in the areas where we lack skills?

Our team has all the necessary expertise and resources, so we do not require any external support or additional team members for the skills assessment. We are fully equipped to handle the required tasks internally.

3.2 Which tools, frameworks, and libraries are most suitable for the project's scope?

Tools:

i. VS Code: An integrated development environment (IDE) for writing, debugging, and managing your Python code.
ii. GitHub: For version control, collaboration, and maintaining code history.
iii. Jupyter Notebooks: Useful for documenting and running Python code interactively, especially in the initial stages of data exploration.

Frameworks:

iv. Scikit-learn: A robust framework for implementing machine learning models and performing tasks like regression, classification, and clustering.
v. TensorFlow or PyTorch: If deep learning is needed, either TensorFlow or PyTorch would be suitable frameworks for building and training neural networks.

Libraries:

vi. Pandas: For data manipulation, cleaning, and exploration. A crucial library for working with structured data.
vii. NumPy: For numerical computations, especially useful in handling large datasets.
viii. Matplotlib/Seaborn: For coding your data visualizations from scratch. You can create detailed and customized plots using Python.
ix. Statsmodels: For statistical modeling and conducting detailed data analysis.

3.3 How can we ensure that each team member is comfortable with the tools selected?

i. Knowledge Sharing: Given that we bring different strengths, set up peer sessions where each member can lead discussions or small tutorials on their areas of expertise. This would allow the team to bridge any knowledge gaps related to dashboards, anomaly detection, or sensor data preprocessing.
ii. Regular Check-ins: Duringteam discussions, ensure that feedback is gathered on tool usage. If anyone is facing challenges, they should be addressed at earliest through extra support or guidance.
iii. Documentation: Create shared documentation or guides for commonly used tasks in the selected tools. This will help everyone stay on the same page and have a reference point.

3.4 Have specific tasks been assigned based on individual strengths, and are team members clear on their roles?

Yes, specific tasks have been assigned based on each team member's strengths, ensuring clarity in roles.
i. Jay, with his experience in building interactive dashboards, is responsible for developing the project's visualization components.
ii. Asmita, given her expertise in anomaly detection and model implementation, is focused on designing and deploying the anomaly detection models.
iii. Jyothssena, with her background in sensor data collection and preprocessing, is handling the data consistency, collection, and preparation tasks. Each member is clear on their role and how it aligns with their expertise, allowing for a streamlined workflow.

4 Initial Setup

4.1 What development environment setup is necessary for this project?

For the project, we will set up the following development environment
i. IDE/Editor: Use VS Code with Python extensions for syntax highlighting and code completion.
ii. Version Control: Implement Git for version control and collaborate via GitHub.
iii. Python Environment: Use Python 3.x and create a virtual environment for managing dependencies.
iv. Libraries: Install essential libraries like numpy, pandas, matplotlib, seaborn, and scikit-learn.

4.2 Have we successfully configured version control (such as Git)? Does everyone have access to the repository?

Yes, we have successfully configured version control using Git, and all team members have access to the repository.

4.3 Have we installed and configured all required software, libraries, and tools?

Yes, we have installed and configured all required software, libraries, and tools for the project.

4.4 What testing can we run to ensure that the development environment is functioning correctly?

i. Model Validation: Use techniques like cross-validation to ensure the machine learning models are performing as expected on the training data.
ii. Unit Tests for Preprocessing: Verify that data preprocessing functions (e.g., normalization, feature extraction) work correctly and produce expected outputs.
iii. Performance Evaluation: Assess model performance using metrics like accuracy, precision, recall, F1-score, and ROC-AUC on validation datasets.
iv. Error Analysis: Analyze misclassified examples or outliers to ensure the model is learning effectively and identify areas for improvement.
v. Dependency Checks: Confirm that all required libraries (e.g., Scikit-learn, NumPy, Pandas) are installed and working correctly.
vi. Environment Consistency Checks: Ensure the development environment matches the specified configurations for libraries and tools.
vii. Reproducibility Tests: Run experiments to verify that results can be reproduced consistently across different runs with the same random seed.

4.5 What troubleshooting steps should we take if the setup does not work as expected?

i. Check Error Messages: Review any error messages or logs generated during execution for clues about what went wrong.
ii. Review Configuration Settings: Double-check configuration files and environment variables to ensure they are correctly set up.
iii. Test Individual Components: Run each component independently to identify which component is causing issues.
iv. Isolate Changes: If recent changes were made, revert to a previous version of the code to see if the issue persists. Use version control tools like Git to manage changes effectively.
v. Check Data Integrity: Verify that the input data is correctly formatted, contains no missing values, and meets the expected specifications.
vi. Consult Documentation: Refer to the documentation for the libraries and tools being used to ensure they are being implemented correctly.
vii. Seek Help from Team Members: Collaborate with team members to brainstorm potential solutions, as they may have encountered similar issues in the past.
viii. Use Debugging Tools: Utilize debugging tools or add print statements to the code to track variable states and identify where the failure occurs.
ix. Recreate the Environment: If the issue persists, consider recreating the development environment from scratch to eliminate any misconfigurations.

5 Progress Review

5.1 What has been achieved so far? Have we completed the initial setup and repository configuration?

We have successfully identified a reliable dataset and started working with it. The initial setup and repository configuration are complete, and we are currently in the process of selecting the models to

use.

5.2 Have there been any issues or blockers, and how can we address them quickly?

So far, we haven't encountered any issues or blockers.

5.3 Is each team member contributing as expected, and does everyone understand their role?

Yes, all team members are fulfilling their responsibilities as expected, and everyone has a clear understanding of their roles.

5.4 Are we on track with the timeline and milestones, or do we need to adjust them?

We are on schedule and confident that we will complete the project within the planned timeframe.

5.5 How does the progress align with the project's overall objectives?

The progress aligns well with the project's goals, and we are making steady progress toward meeting the objectives.

6 Plan Revision

6.1 Based on progress so far, do we need to adjust the project timeline or milestones?

Although we are currently on track, we can conduct a mid-point review of our timeline and milestones to ensure that we stay ahead of any potential delays and make adjustments if needed.

6.2 Are any tasks delayed or requiring reassignment due to workload or skill gaps?

No tasks have been delayed so far, but we will continue to monitor individual workloads and skill needs, so if any issues arise, we can quickly reassign tasks or provide support where necessary.

6.3 How can we ensure that all members are clear on the revised plan and their next steps?

In addition to a team meeting, we can share a detailed written breakdown of the revised plan with clear deadlines and responsibilities. This can be supplemented with a shared task board to ensure everyone remains aligned on their next steps.

6.4 What communication strategies can we implement to avoid future delays or misunderstandings?

Along with regular check-ins, we could assign a team member to oversee communication flow, ensuring that important updates are shared promptly and any questions are addressed. Establishing clear communication channels, like Slack or Trello, will also streamline discussions and task management.

6.5 How will we track progress going forward and maintain alignment with the revised plan?

We will maintain alignment by using a project management tool (e.g., Jira or Asana) that tracks the progress of each task, assigns deadlines, and allows the team to visualize progress. Weekly progress reviews can help us identify any bottlenecks and adjust as needed to stay on track.

7 Submission for This Iteration

7.1 If your data is available online, please provide a link to access it.

For this iteration's submission, document the project kickoff by clearly defining the goals, scope, deliverables, milestones, and dataset selection. Include team discussions about individual skills and task assignments. Assess tools and frameworks, ensuring comfort with the selected platforms. Confirm initial setup, repository configuration, and progress alignment with the objectives. Lastly, provide the GitHub repository link and confirm that the PDF reflects the team's actual progress.

7.2 Have we detailed the challenges faced, the solutions implemented, and any adjustments to the plan?

Since this is the initial stage of the project and the team hasn't yet faced significant challenges or implemented any solutions, there are no major adjustments to report at this point. The team has successfully completed the pre-documentation, set up the development environment, and established roles, but challenges and adjustments will be documented in future iterations as the project progresses.

7.3 If your data is available online, please provide a link to access it.

No, dataset isn't available online.

7.4 Is the PDF using the Overleaf template, and does it reflect the team's actual progress?

Yes, the PDF is using the Overleaf template and yes, it reflects the team's actual progress.

7.5 Does the submission meet all the project requirements, and is it ready for review by stakeholders?

Yes, it's ready to be reviewed by stakeholders.

7.6 GitHub repository Link

https://github.com/JayJajoo/IDMP-project.git