CS-359 Machine Learning

Architectural Decision Record and Final Report

What is an ADR?

Michael Nygard, a pioneer in the field, outlined the necessity for ADRs in 2011. These records serve as a repository for the rationale and justification of every significant decision in an engineering project. Data and machine learning architects/engineers should also identify and analyze potential consequences, advantages, and disadvantages.

Why ADRs are necessary?

ADRs are primarily valuable for communicating initial decisions, discussions, and updates on every significant step in a project. It is essential to highlight the consequences and different opinions on every decision made by the team. Stating the team members' names and their views on every decision is helpful for sustaining the project.

ADR of a Machine Learning Project

Any project can have a long list of decisions and used technologies. I will outline the essential stages of a machine learning project, including the data engineering and machine learning pipeline. You may add other decisions as you see fit, but removing any of the following elements should be avoided.

1. Executive Summary and Project Goals

Write an executive summary of the project in your own words for around 1,000 words. People with minimal domain knowledge should understand the summary.

2. Data Sources

2.1 Context and Technology Choice

Identify the data sources used in the project.

2.2 Justifications

Justify your choices in .1

2.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

Data Engineering Pipeline - Extract, Transform, Load (ETL)

3. Data integration (extract)

3.1 Context and Technology Choice

Data comes from different sources, including web sources, social media, streaming devices, structured sources such as relational databases, and semi-structured sources such as e-mails, JSON, XML files, or NoSQL databases. This stage aims to integrate the data from multiple sources and prepare it for transformation and loading.

Identify the data sources, the technology used to ingest them, and any staging areas created to store the data for transformation temporarily. Staging areas could be a database or simply a Pandas data frame.

3.2 Justifications

Justify your choices in .1.

3.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

4. Data Transformation (Transform)

4.1 Context and Technology Choice

Data transformation tasks in the data engineering pipeline should be discussed and agreed upon by the data engineering and machine learning teams. The transformation tasks should be minimal, and purely machine learning and model-dependent tasks should be left to the feature engineering task. Tasks may include:

- Data types conversion

- Data format conversion (cm to inches, etc.)
- Remove duplicates
- Identifying errors in data
- Handling out-of-range and outlier data
- Add any other transformations you find necessary.

4.2 Justifications

Justify your choices in .1.

4.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

5. Data Storage (Load)

5.1 Context and Technology Choice

After performing the necessary transformations, data should be stored in storage mediums/devices and ready for machine learning engineers to consume. Storage could be a simple CSV or JSON file, SQLite database, MySQL, or PostgreSQL database. On a larger scale, storage systems may include warehouses, data lakes, and data marts.

5.2 Justifications

Justify your choices in .1.

5.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

6. Reading Data

6.1 Context and Technology Choice

Read the loaded data in the ETL load stage into a Pandas data frame or any other data structure you choose.

6.2 Justifications

Justify your choices in .1.

6.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

7. Exploratory Data Analysis

7.1 Context and Technology Choice

Exploratory data analysis (EDA) is essential to understanding the dataset and making informative decisions regarding model selection or feature engineering. EDA may include the following:

- Identifying the shape of the dataset
- Unify the columns/features names
- Identifying the unique values in the class label variable
- Identifying if the dataset has missing data
- Identifying columns with a high percentage of missing data
- Performing univariate analysis
    - Statistics of columns: mean, standard deviation, or variance. Identify columns with low or near-zero variance. These features/columns carry little information and may be removed in the feature engineering stage.
    - Visualization: bar plots, pie chart, boxplot, or violin plot
- Performing bivariate analysis
    - Generating pair plots of each pair of columns
    - Computing and visualizing the correlation matrix of the dataset columns or a partial subset of them.
- Performing multivariate analysis
    - Perform cluster analysis (K-means or DBSCAN/HDBSCAN) on a selected subset of features.
    - You may set the number of clusters to the number of classes in the label vector.

    - You may check if the predicted cluster index correlates with the class index.
- Identify outliers
    - Identify outliers using interquartile ratio (IQR): IQR = Q3 – Q1, with data with values less than (Q1 – 1.5*IQR) and (Q3 + 1.5*IQR) are considered outliers.
    - Identifying outliers using histograms or boxplot

7.2 Justifications

Justify your choices in .1.

7.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

Machine Learning Pipeline

8. Data Preprocessing

8.1 Context and Technology Choice

Techniques include (in this order; refer to data leakage slides for details):

- Duplicate removal
- Splitting the dataset into train and test splits (and a validation split, if needed)
- Handling missing data using data deletion or imputation. Ensure that test statistics are never used for data imputation.
- Encoding categorical data using label encoding or one-hot encoding.
- Add any other techniques you find needed.

8.2 Justifications

Justify your choices in .1.

8.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

9. Feature Engineering

9.1 Context and Technology Choice

Feature engineering consists of selecting, creating, or transforming features to improve the model performance or reduce computations. Techniques include:

Feature Scaling

Techniques may include:

- Scaling into a specific range, for instance [0, 1]
- Normalization (feature std is scaled to 1) or standardization (feature mean=0 and std=1)

Feature Selection

Techniques may include:

- Remove features with near-zero variance.
- Remove one feature from each pair of correlated features.
- Perform forward, backward, and recursive feature selection.
- Filter-based methods.
- Perform LASSO feature selection.
- Perform feature selection using feature importance in tree-based classifiers/regressors (XGBoost or Random Forest).

Feature extraction

Techniques may include:

- Principal component analysis (PCA)
- Linear discriminant analysis (LDA)

9.2 Justifications

Justify your choices in .1.

9.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

10. Processed Data Loading

10.1 Context and Technology Choice

Load the cleaned data in step 9 into a storage medium (a file or a database).

10.2 Justifications

Justify your choices in .1.

10.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

11. Model Selection and Training

11.1 Context and Technology Choice

Select the model with the best performance among the options available to solve a given problem. You can use any classification algorithm, including those covered in the course. Train the models using the training dataset. You should compare the performance of at least five classification algorithms.

11.2 Justifications

Justify your choices in .1.

11.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

12. Model Evaluation

12.1 Context and Technology Choice

The model is evaluated using a test dataset it has not seen before. Evaluation metrics:

- Classification: Accuracy, precision, F1-score, confusion matrix, and AUC.
- Regression: mean-squared error, mean absolute error, coefficient of determination (R2).

Model evaluation and selection is an iterative process.

12.2 Justifications

Justify your choices in .1.

12.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.

13. Model Deployment and Monitoring (extra task)

13.1 Context and Technology Choice

The trained model is made available for clients to perform the designed tasks. Refer to the chapter "Embedding a machine learning model into a web application " in the book "S. Raschka and V. Mirjalili, Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow. Packt Publishing.

13.2 Justifications

Justify your choices in .1.

13.3 Status

State the status of this decision, as well as names and dates. The status could be proposed, accepted, or rejected.