**Assignment – Trendence**

**Feature Store:**

A feature store provides a single pane of glass for sharing all available features. When a data scientist starts a new project, he or she can go to this catalogue and easily find the features they are looking for. But a feature store is not only a data layer, it is also a data transformation service enabling users to manipulate raw data and store it as features ready to be used by any machine learning model.

**Need or motivation for feature store in Machine Learning:**

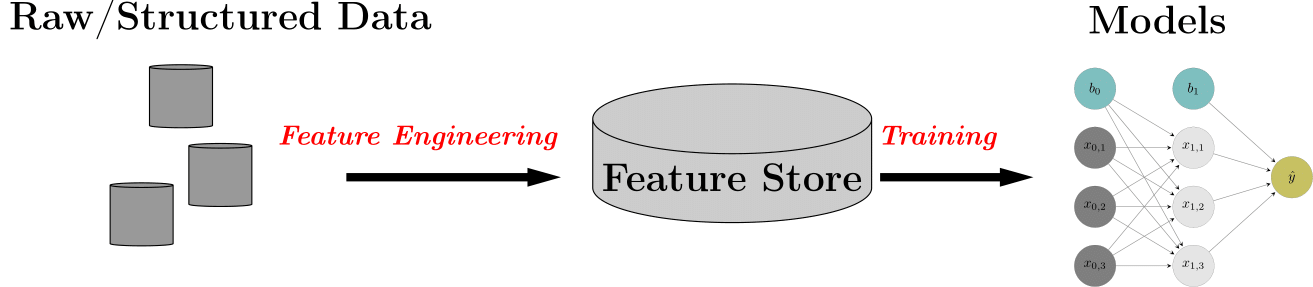
* There is no principled way to access features during model serving.
* Features cannot easily be re-used between multiple machine learning pipelines. Data science projects work in isolation without collaboration and re-use.
* Features used for training and serving are inconsistent.
* When new data arrives, there is no way to pin down exactly which features need to be recomputed, rather the entire pipeline needs to be run to update features.

**Used for or problems it solves:**

Using a feature store is a best practice that can reduce the technical debt of machine learning work-flows. When the feature store is built up with more features, it becomes easier and cheaper to build new models as the new models can re-use features that exist in the feature store. The Feature Store provides, among other things:

* Feature reuse
* Feature discoverability
* Feature backfilling and precomputation
* Improved documentation and analysis of features
* Software engineering principles applied to machine learning features: versioning, documentation, access-control
* Scale; The feature store needs to be able to store and manage huge feature sets (multi- terabyte at least).
* Flexibility; Data scientists must be able to read from the feature store and use the data in different machine learning frameworks, like Tensorflow, Keras, Scikit learn, and PyTorch. Analysis; Data scientists need an understanding of the feature data to be able to make most use of it in their models. They should be able to analyze the features, view their distributions over time, their correlations with each other etc.
* Point-in-time correctness; It can be valuable to be able to extract the value of a feature at a specific point-in-time to be able to later on change the value of the feature.
* Real-time; for client-facing models, features must be available in real-time (< 10ms) for making predictions to avoid destroying the user-experience for the user.
* Online Consistency; when a feature is used for both training and serving, and stored in two different storage layers, you want to make sure that the value and semantics of the feature is consistent.

**Feature store:**



* The **feature** is an individual versioned and documented data column in the feature store, e.g the average rating of a customer.
* The **feature group** is a documented and versioned group of features stored as a Hive table. The feature group is linked to a specific Spark/Numpy/Pandas job that takes in raw data and outputs the computed features.
* The **training dataset** is a versioned and managed dataset of features and labels (potentially from multiple different feature groups). Training datasets are stored in HopsFS as tfrecords, parquet, csv, or tsv files.

