Building and Deploying a Multi-Stage Recommender System with Merlin

Karl Higley, Even Oldridge, Ronay Ak, Sara Rabhi, Gabriel de Souza Pereira Moreira

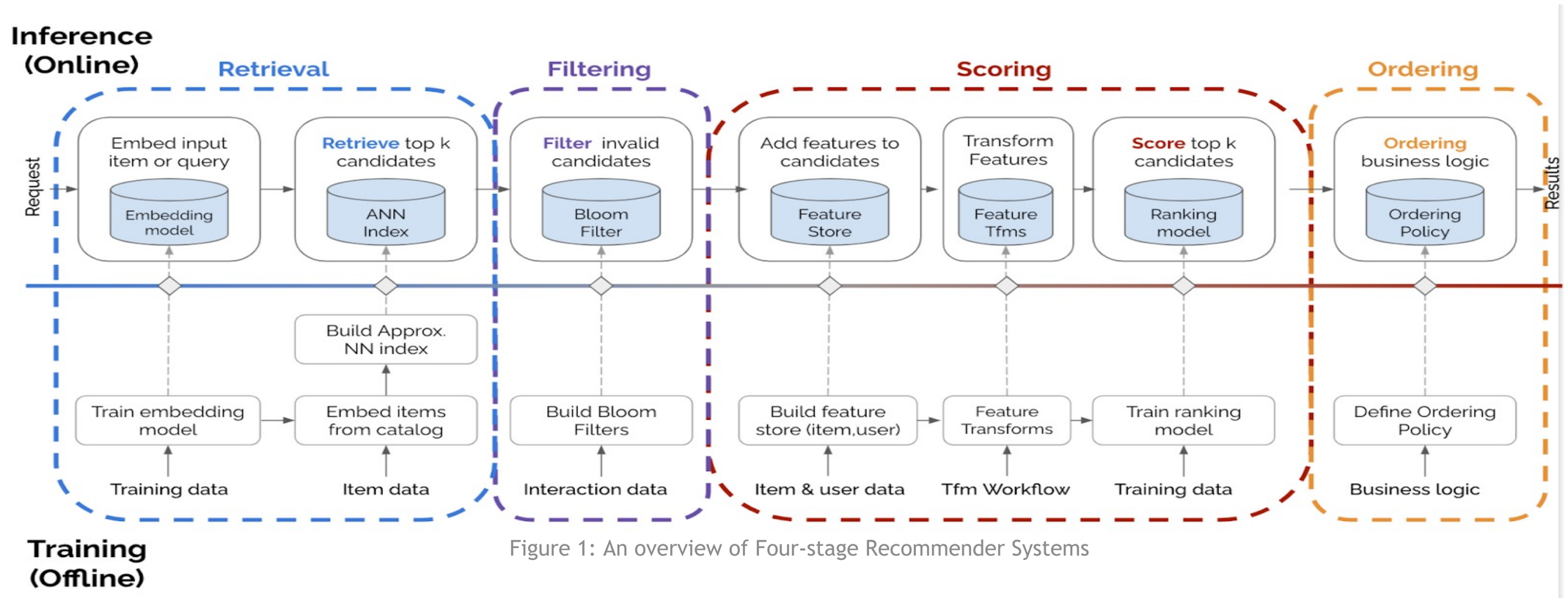

Figure 1: An overview of Four-stage Recommender Systems

1. Introduction & Motivation

- A gap exists between open-source research-focused frameworks and production systems that serve online recommendations.
- Recommender System includes different inter-connected components beyond the models and algorithms.
- Scalability, performance, integration of business rules for filtering and ordering should be considered when building a recommender system for production usage.

Summary of our main contributions for multi-stage recommendation system

- Identify and formulate a common pattern of how production recommender systems look like, which we call four-stage recommender systems.
- Develop an open-source library, Merlin Systems, to leverage the necessary assets for building a multi-stage system and deploying such systems ensemble into a Triton Inference Server
- Present a demo of a multi-stage recommender system use-case

2. Four-stage Recommender Systems

- The four stages of Retrieval, Filtering, Scoring, and Ordering is a design pattern that covers most of the existing recommender systems and is presented in Figure 1:

1. **Retrieval:** In practice, the items catalog is very large (millions to billions) and applying the scoring model to the whole set is not feasible. The retrieval models are lightweight and scalable models that sub-select a smaller set of relevant candidates (a thousand to ten thousand).
2. **Filtering:** Exclude the already interacted or undesirable items from the candidate items set or to apply business logic rules such as remove out of stock products.
3. **Scoring**: This is also known as ranking. Here, we apply the complex ranking model to score the level of interest that a user may have in each item of the filtered candidates set.
2. **Ordering**: Order the final set of items that we want to recommend to the user. Here, we are able to align the output of the model with business needs, constraints, or criteria.

3. End-to-end example of a multi-stage recommender system

The first part of the demo focuses on the offline training and the second part details how to deploy such multi-stage system into production.

Offline training of the multi-stage components

1. Train a retrieval model (Two-tower architecture) using TensorFlow
2. Train a ranking model (DLRM) using TensorFlow
3. Export unique users and unique item features and load into a Feature Store (Feast)
4. Export the user tower of the trained ranking model for inference
5. Use the retrieval item tower to generate and persist the unique item embeddings
6. Index the item embeddings into an ANN index (Faiss)

Online deployment of the multi-stage system

Merlin Systems can be used to connect the offline assets, build a Directed Acyclic Graph (DAG) and deploy such models ensemble into a Triton Inference Server. In Figure 2, you can see the flow which starts with a request with a user-id and finishes returning the Top-k recommended items for this given query.

4. Conclusion

- Illustrate the multi-stage system, a design pattern including the most common steps found in large companies' recsys pipelines.
- Introduce the Merlin framework, which supports all necessary components to preprocess historical data, define and train retrieval and ranking models, apply filtering and re-ordering operators, and prepare feature stores and ANN index for online serving.

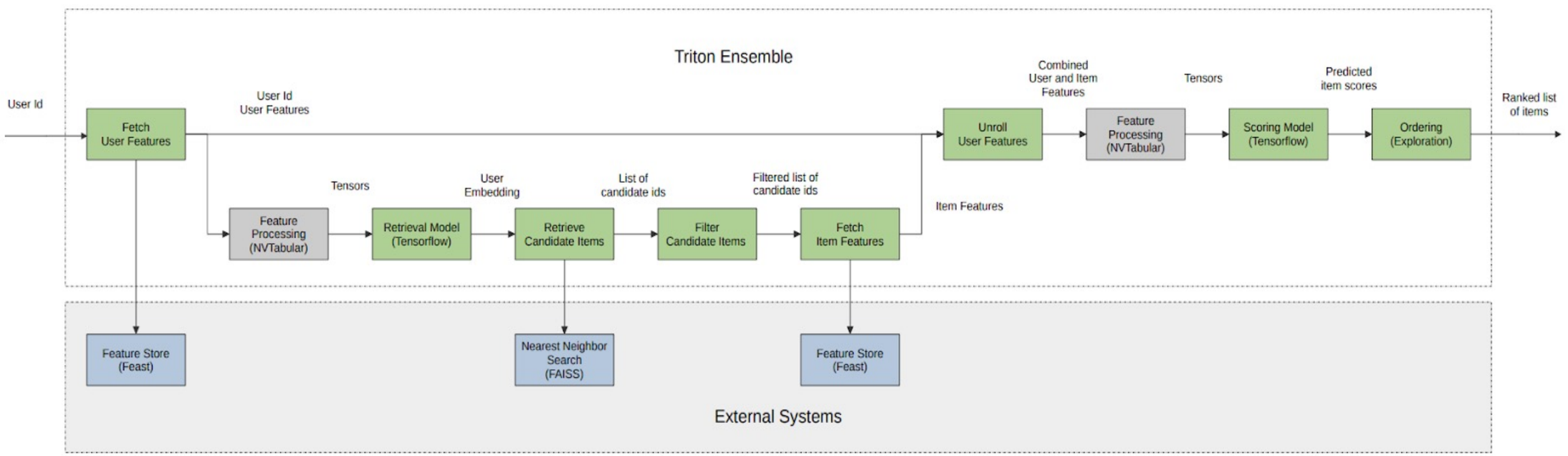

Figure 2: A Four-stage Recommender Systems inference pipeline using Merlin and Triton Inference Server