# Demo for a Recommender System that Can Efficiently "Forget" User Data

Hengyu Tang, Shirley Xu, Nhung Le, Enyi Lian
ht1162, xx852, nhl256, el2986@nyu.edu

December 12, 2019

# Overview

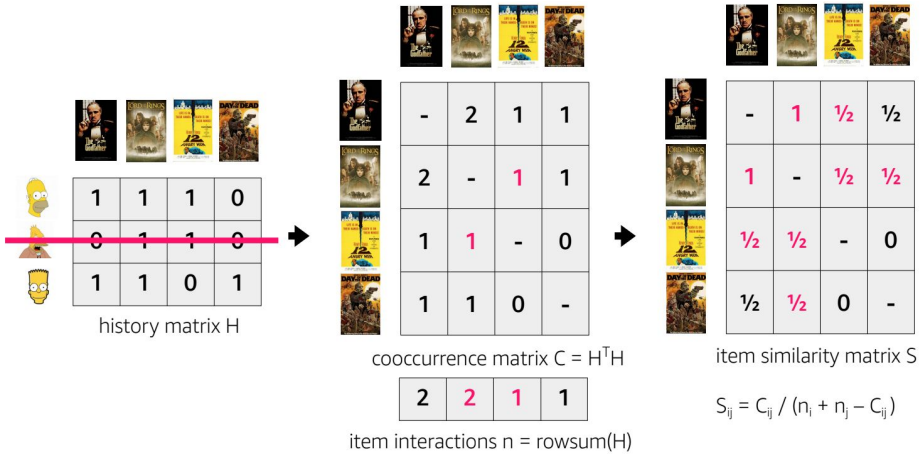
- Problem Statement and Approach
- Implementations
- Demo
- Discussion

# Problem Statement

- **Motivation**
  - EU passing laws (GDPR) to address privacy issues
  - GDPR requires the deletion of personal data upon request
- **Amnesia**
  - **Decremental learning process** allows trained ML models to “forget” user data **efficiently** and **reliably**
- **Goal**
  - Create a WebApp that allows user interactions and demonstrates the decremental learning of *Amnesia*

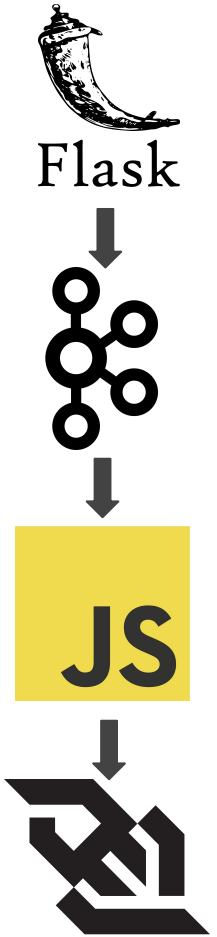
# Approaches

- **Data**: static input → user live input
- **Expected Output**
  1. History matrix
  2. Total item interactions
  3. Co-occurrence matrix
  4. Similarity matrix
- **Tools**
  1. Python Flask library + Kafka API
  2. JavaScript + WebSocket API
  3. HTML + CSS



# Implementation

- **Stage 1: Matrices Load & Print**
  - **JQuery** animations to show changes
- **Stage 2: Real-time Updates**
  - **Apache Kafka** to connect to the model
  - Session for data storage
- **Stage 3: User Interactions**
  - Allow audiences to **remove** or **add** user
  - Cater to edge cases
- **Stage 4: Optimizations**
  - Multithreading + subprocess
  - Reduce waiting time to present web updates, resort to **WebSocket API**



# Demo

# Discussion

- Observations
  - Limits of python based applications
  - Flexibility and fast-processing of JavaScript and WebSocket API

- Future Topics
  - Deprecation and limitations in **python-kafka** package and potential fix-up
  - Integration of k-means clustering and recommendations to present the “forget” effects on warm-start problem
  - Creating a user interface is easy; creating a good one is very hard