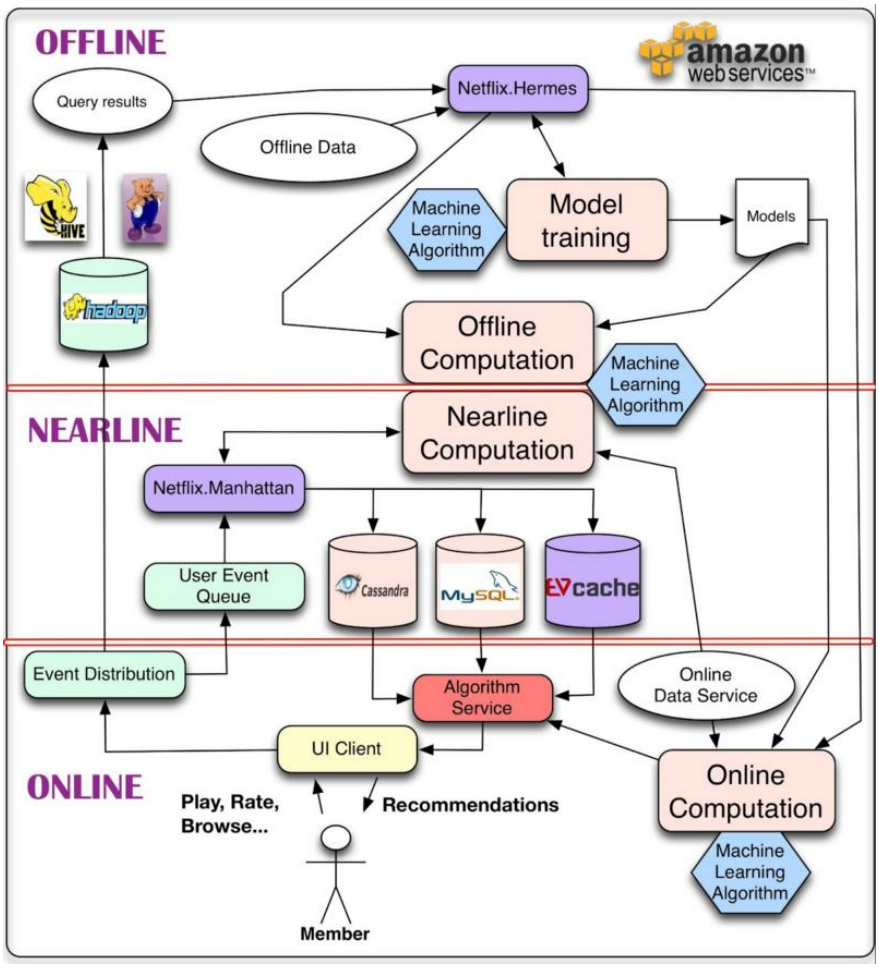
System Architectures for Personalization and Recommendation

https://medium.com/netflix-techblog/system-architectures-for-personalization-and-recommendation-e081aa94b5d8



처음에는 전체적인 추천시스템의 전체적인 diagram을 살펴보도록 하겠다.

- 우선 나중에 offline processing 할 수 있도록 데이터를 처리해 준다.
- Online computation의 경우 최근 이벤트와 유저간의 상호작용을 좀더 좋게 반응 할 수 있다. 그러나 실시간 요청에 반응해야 되기에 데이터를 처리하는 알고리즘의 계산상의 복잡성에 한계가 있다.
- Offline computation은 알고리즘의 계산상의 복잡성과 데이터의 양의 한계가 좀더 적지만 최신의 데이터와 결합을 하지 않기에 업데이트 사이에 오래된 데이터가 될 수 있다.

****One of the key issues in a personalization architecture is how to combine and manage online and offline computation in a seamless manner.****

- Nearline computation은 Online과 Offline computation의 중간 타협점이라고 생각하면 된다.
- Model training은 존재하는 데이터를 이용해 실제 결과 계산 중에 사용될 모델을 생성하는 연산이다.

Online, Offline, Nearline computation

Online computation
-최근 event에 반응하며, 가장 최신의 data를 이용한다.
-모델을 fit하기 어려우며 계산상 비용이 많이 드는 알고리즘이다.
Ex) Service Level Agreements

Offline computation
-사용하는 데이터의 양이 제한적이다.
- engineering 요구사항은 단순하다. 고객들이 요구하는 것들에게 대해서 잘 처리할 수 있다.
- 새로운 데이터나 바뀐 context에 대해선 빠르게 반응하지 못한다.
Ex) Movies

Nearline computation
-Online과 Offline을 절충한 것이다.
-Computation는 정확히 말해 online을 방식을 이용한다.
-결과가 계산 되는 즉시, 요구사항들을 지워버린다.
-복잡한 처리가 가능한 이벤트에 대해 열러 있다
Ex) immediate watching movie

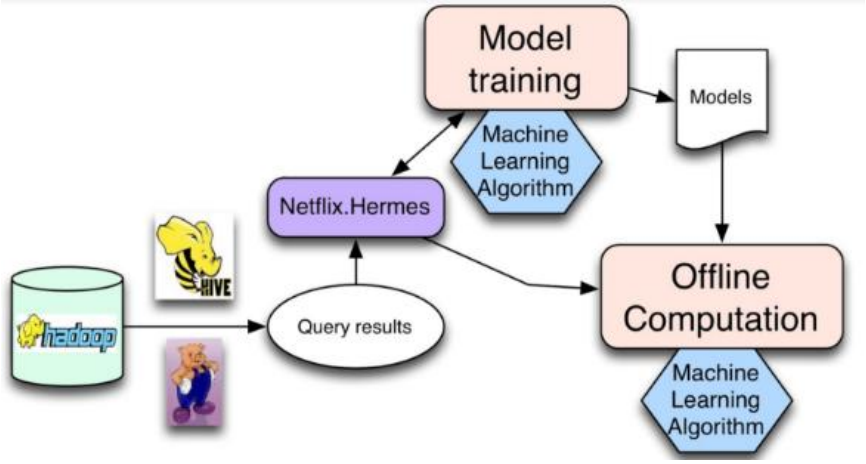
3가지 방법들은 결합해서 사용하는 방법은 여러 가지가 있다.

모델링 part에서는 hybrid offline/online 방식으로 사용할 수 있다.
대표적으로 matrix factorization이 있다. 일부 요인은 오프라인으로 사전 계산하고, 다른 요소는 실시간으로 업데이트하여 새로운 결과를 만들어낸다.

비 지도학습인 클러스터링은 클러스터계산을 offline으로 실시하고 클러스터를 온라인으로 assignment한다. 즉 크고 잠재적 복잡한 global model을 training하는 부분과 좀더 가벼운 user-specific model을 training 하는 부분으로 나눌 수 있다

.

Offline jobs



1. model training

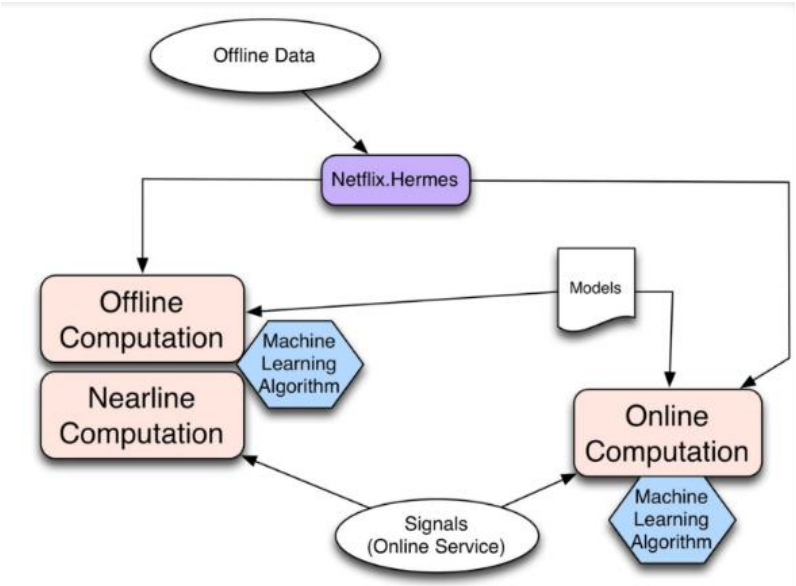
-존재하는 데이터에서 연관성을 찾고 머신러닝 알고리즘을 이용하여 모델의 parameter를 생성한다.

2. batch computation of intermediate or final result

- 대부분의 모델들은 offline으로 일괄처리 방식으로 train하지만 online 상에서 incremental training 방법을 통해 online 학습도 한다.

- batch computation result는 가지고 있는 모델과 input data를 사용하여 결과(후속 online processing or direct presentation to the user)를 산출하는 offline computation process이다.

Signals & Models



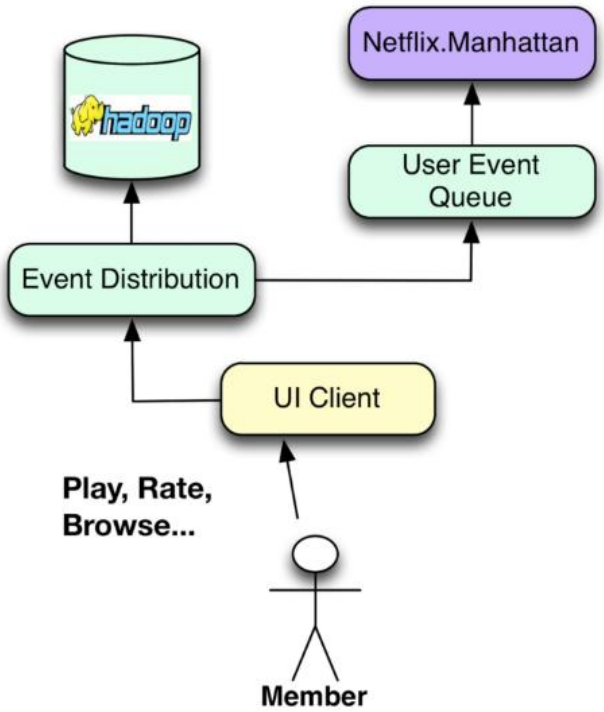
Online or Offline으로 계산하든지 간에, 알고리즘이 3가지 입력(Model, data, signals)을 어떻게 처리할지 생각해 봐야 한다.

Model – 이전에 offline으로 훈련된 작은 parameter 파일이다

Data – 데이터베이스에 저장된 이전에 정보이다(ex- movie metadata and popularity)

Signals – 실시간 서비스에서 얻을 수 있는 데이터, user-related information을 만들 수 있다.
EX) 최근에 무엇을 보았는지, Context data(session, device, date, time)

Event & Data Distibution



-우리의 목표는 member interaction data를 이용하여 member`s experience를 향상시키는 것이다.
-Clicks, browsing, viewing 등의 event데이터들을 통합하여 알고리즘을 만들어간다
-event는 최소 지연시간으로 처리해야 되는 정보의 가장 작은 단위이며 nearline result set과 같은 후속 action이나 process을 trigger한다.
-data는 나중에 저장하고 사용하기 위해서 좀더 dense한 information이라고 생각한다.