**1 Question 1 (5 Points / 12 Minutes)**

**What is a Recommender System? Describe the purpose of the system and list its**

**necessary components.**

Recommend systems that recommend items to users based on which items are thought to be a good preference match for the user. This can be done in many ways but there are 4 main ways of accomplishing this.

1. Collaborative filtering:
   1. Items-Items: “People who liked this item also liked these items”
   2. User-Items: “Users like you also like these items”
2. Content based filtering: “This item you like is similar to these items”
3. Knowledge based filtering: Rule based systems. “If age = underage, recommend pg < 18”.
4. Hybrid: combination of the three other methods

The necessary components are users, items, objective function and recommendation algorithm. Other aspects include evaluation, user interface, contextual information, behavior etc.

**2 Question 2 (5 Points / 12 Minutes)**

**In Content-based Filtering (CBF), we estimate users’ preferences toward items’ features.**

1. **List three methods to represent items in terms of their features. (1.5 points)**

Bag of words (BoW): Each item is represented with a word count based on a fixed vocabulary. You could optional represent user preferences as an average vector of the items they liked. Each dimension corresponds to a word in the vocabulary.

TF-IDF(term, document): = TF(term, document) \* log(1/DF(term)), reduces the impact of frequent words like “the”, “is” etc and helps prioritize rare words like spaceship, vampire, heist etc. This improves upon BoW by reducing the impact of common words.

Word Embeddings: These are word embeddings where each word has a multi-dimensional dense vector. Closer words are more related and words farther apart have little to no relation. Captures semantic meaning of item content.

1. **Suppose we have item representations x and want to estimate the rating for an unobserved (user, item) pair. Describe how you would compute the rating score.**

Apple 1 0

Geof 0 1

First we defin the user’s interaction set denoted by J\_i. J\_i is the interaction set performed by user i. Then we represent items with features denoted by X\_j which is a feature vector. After that we represent the user profile X\_i. This is done using an aggregator function F.

X\_i = F(X\_j : j E J\_i), This just means that we will aggregate the user profile using X\_j (the set of item features) where j belongs to the set of J\_i (items that the user has interacted with.) The aggregation function could be weighted average where we multiply (X\_j : j E J\_i) with the rating the user gave for each of these items and divide it by the sum of the ratings. This will normally give us a normalized result but if not, depending on the representation of item features, normalize the results. After we have X\_i (user profile) we then compute a similarity to the new item using methods like cosine similarity (mostly used, also keeps angle information), Manhattan distance, KNN etc. For cosine similarity, the score would be the dot product of the item and user feature vectors divided by the product of L2 norms of both. This makes it so that the similarity focuses only on the angle between them and not their magnitude (long/short vectors). The best score would be 1 where the vectors are parallel to each other and pointing in the same direction. This is unsupervised scoring where item features and user features are in the same vector space.

Supervised scoring on the other hand learns from observed ratings to predict ratings for user item pairs based on their feature. A simple example can be a regression model

**Collaborative Filtering (CF) has been established as one of the most successful recommendation methods.**

1. **Describe the process of user-based collaborative filtering with the nearest neighbor approach.**

Collaborative filtering bases itself on the idea that similar users like similar items (groups of users have preferences for a shared subset of items). How do we represent preferences? We have a matrix Yij that represents the rating of user I towards item j. MxN where m is the set of users and N is the set of items.