1.4:Terry Phung

In order for the recommender system to do its task of identifying the most relevant item for the user, the recommender must predict that an item is the closest to what the user wants. To predict the most relatable item to what the user wants, the recommender system employs several techniques to determine the most relevant item. The first of these techniques is a content-based algorithm that recommends items based on what the user liked in the past. For example, if a user played and liked an action game in the past, then the system would recommend an action game that's similar feature wise. The second technique is a collaborative filtering algorithm that doesn't just recommend items based on the similarity of its features to an item that a user used before, it recommends items based on the ratings of users with similar tastes too. Using the example from the content based algorithm, collaborative filtering recommends a game that is similar feature wise to a game a user played before and by the reviews of other users. The third technique is a demographic algorithm where instead of recommending items based on the similarity of items and user reviews, the demographic algorithm recommends items based on a demographic's assumed preferences. For example, if a user is part of the "children" demographic, the algorithm will recommend items that it deems appropriate to that demographic like toys or E rated movies. The fourth technique is a knowledge-based algorithm that recommends an item based on the explicit preferences of its users. For example, if the user inputted directly what they want, let's say a car that cost around the $2000-5000 range, then the system will recommend items with that specific filter in mind. The fifth technique is a community based algorithm that works similarly to a collaborative filtering algorithm but instead of depending on anonymous user ratings it depends on the ratings of the user's friends. The algorithm assumes that the user will likely choose an item based on the recommendation of their friends. The last technique that the author discussed is a hybrid recommender algorithm which takes a combination of features of all the techniques that was discussed. This algorithm can vary, meaning a hybrid algorithm can have features from a demographic algorithm and a content filtering algorithm or any mixture of techniques.

1.5 and 1.6: Christopher Wu

Recommender systems(rs) are now mostly used commercially, and the different applications of a recommender system will affect the design of the system, its implementation, maintenance and enhancement. The first part of the design to consider is the domain, or what context will the rs be used in. The developer should take into account the specifics of the domain, the requirements, challenges, and limits before choosing which recommender algorithm to choose. A three approach model that can be used is thinking about system users, the characteristics of the data, and the overall application.

Another issue to think about while determining effectiveness of a recommender system is evaluating how accurate the system is giving recommendations. In the design phase, the recommendations the algorithms give should be compared to real-world data, like user interactions or ratings. When the system is launched, evaluation should be done again to make sure the output is similar to what users want. The way to determine that is compare real users' ratings with what the algorithms normally output. If the results are very different, then different weights should be adjusted to make the data align.

In a perfect world, recommender systems would only give the best options to users and they would accept it, but that does not happen. The whole point of recommender systems is that the user is not informed enough to make a decision on their own, but it also means that they are not informed to know what a good recommendation is. The effectiveness of a rs is not just how good the recommendations are, but also how much trust the user has in the recommendations and how likely they are to use it. The rs must inspire trust with a user with a variety of techniques: have logic that is transparent, show new items, provide details about the recommended item, and give ways to refine the recommendations.

Trust is built on two different fronts: trust in other users of the recommender, and trust in the system. One way to build trust in the system is by leaning into the trust between users. For example, in social recommender systems, such as Facebook and LinkedIn, users are more likely to trust the system if it recommends other users that they trust, such as coworkers or friends. Some other ways to build trust with a user is transparency and scrutability. There should be an explanation of why the system gave out the certain recommendations. For example, the system could say “item x was recommended because of feature a and feature b” to show the user why the recommendations were chosen. One limitation of recommender systems is that the algorithms run once and do not take into account the feedback given by the user. A solution to this problem is using conversational rs. They are built on having an interactive process between the user and the system.

Karnendra Verma

1.7+1.8

Recommender Systems are multidisciplinary in nature; for example they draw from machine learning , information retrieval , and user psychology. Machine learning and Data mining draw from AI concepts to efficiently perform tasks based on data, data usage and constraints , as well as past user experiences. We can also use them to model and predict future performance within certain paradigms. We construct these through various statistical models such as regressions. Keep in mind though, these predictions are not 100 percent accurate, though they can have a high utility given a sufficiently accurate model. Recommender Systems utilize machine learning and data mining techniques along with filtering, ranking, processed feature weighting, and incorporation of weighted feature sets. A few examples of the data mining techniques Recommender Systems typically take advantage of are data preprocessing methods such as sampling and dimensionality, support vector machines, as well as clustering K-means, hierarchical clustering (top down bottom up), and association rule mining. Association rule mining is a technique to define a collection of items or variables between datasets. Two common algorithms are APRIORI and SETM .Common applications are for categorizing abstract data as well as modeling emergent behaviors within the datasets. We may use this in our project to see what types of workouts a user may like and try to link it to a group or subgroup of users. Both Recommender Systems and Information Retrieval face overlapping problems in regards to filtering and ranking. IR's can oftentimes overlook individual needs and "unique interests" as well as usefulness. Further, it is possible that as a feedback loop over time, we may end up with more sameness and less diversity within a customer base, or even a population.

When examining the future of Recommender Systems, we see a trend towards social gamification. When you visit a new restaurant and Google, it will ask for a review, rating, and likely aggregate your preferences into a profile of "what you like" and what other people like you like. This has real world consequences, and businesses today can thrive or die by these rankings and ratings. In theory, given enough robustness a system would work 100% of the time in minimizing fake reviews or profiles , but there is money to be made. So much so that even Amazon and other multi-billion or trillion dollar companies have not found ways to make their Reccomenders infallible to bots that rate products favorably or write fake reviews. Privacy concerns are also ever present, many of the best and most profitable Reccomenders use enormous data sets, and that means that data has to be collected, stored and organized in a way that makes Data Retrieval quick and efficient. This exposes risk in terms of Cybersecurity and customer personal identifying data.