**CB, CF Challenges**

1. Cold Start Problems

The cold start problem occurs when the system is unable to form any relation between users and items for which it has insufficient data. The recommender systems face a problem in recommending items to users in case there is very little data available related to the user or item.

1. New user problem: there is very less information about the user
2. New item problem: When an item added to the catalogue has either none or very few interactions.  for [collaborative filtering](https://en.wikipedia.org/wiki/Collaborative_filtering) algorithms due to the fact that they rely on the item's interactions to make recommendations. If no interactions are available then a pure collaborative algorithm cannot recommend the item. In case only a few interactions are available, although a collaborative algorithm will be able to recommend it, the quality of those recommendations will be poor.

[Content-based filtering](https://en.wikipedia.org/wiki/Content-based_filtering) algorithms, on the other hand, are in theory much less prone to the new item problem. Since content-based recommenders choose which items to recommend based on the feature the items possess, even if no interaction for a new item exists, still its features will allow for a recommendation to be made.[[7]](https://en.wikipedia.org/wiki/Cold_start_(recommender_systems)#cite_note-Pazzani07-7) This of course assumes that a new item will be already described by its attributes, which is not always the case. Consider the case of so-called *editorial* features (e.g. director, cast, title, year), those are always known when the item, in this case, a movie is added to the catalogue. However, other kinds of attributes might not be e.g. features extracted from user reviews and tags.[[8]](https://en.wikipedia.org/wiki/Cold_start_(recommender_systems)#cite_note-8) Content-based algorithms relying on user-provided features suffer from the cold-start item problem as well, since for new items if no (or very few) interactions exist, also no (or very few) user reviews and tags will be available.

1. New system: startup of the system when virtually no information the recommender can rely upon is present.
2. Sparsity Problem: This occurs Most users do not rate most of the items. Recommenders use large datasets. Therefore, the user-item matrix used for filtering could be very large and sparse and because of that performance of the recommendation process may get degraded.

Collaborative filtering suffers from this problem because it is dependent on the rating matrix in most cases.

1. Data Scalability: The amount of data used as input to RS is growing quickly as more users and items are added. Despite a large amount of data, most RS aspire to respond interactively in less than a second in order to keep users engaged. A key challenge here is to design efficient learning algorithms that can handle such large-scale datasets.

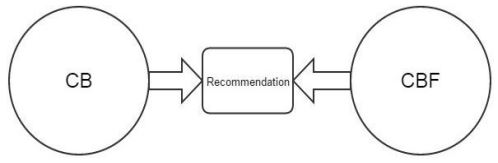
**Hybrid RS**

Hybrid recommenders are systems that combine multiple recommendation techniques to improve recommendation accuracy.

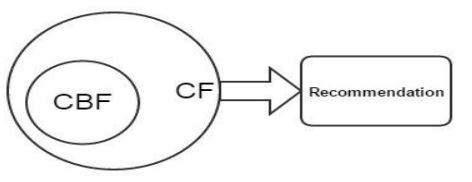
Most popular collaborative filtering and content-based filtering.

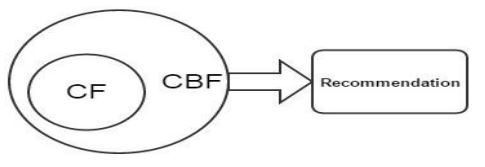
Hybrid approaches can be implemented in various ways

1. Implement collaborative and content-based methods individually and aggregate their predictions



1. Integrate some content-based characteristics into a collaborative approach, or vice versa.





1. Construct a general consolidative model that integrates both content-based and collaborative characteristics

