Collaborative Recommendation - User Similarity
Content-Based Recommendation - Item Similarity
Knowledge-Based Recommendation - User Interaction / Domain Knowledge

- Example: Constraint-Based Recommenders

Hybridization - Collaborative & Content-Based

- Question: Why not include Knowledge-Based Recommendation? Because when CR and CBR share similar learning and improving methodology, KBR more focus on filtering and restricting to correct or further improve hybridization recommender's output. (i.e. different level in the system)

Collaborative Recommendation

- User-Based Nearest Neighbor Recommendation
    - Similarity Measure: Pearson Correlation Coefficient - Range [-1, +1]
        - Best Measure for User-Based Recommender System - Demonstrated by Empirical Analysis
        - Potential Problem: Discriminativeness & Popularity
            - Solution: Inverse User Frequency / Variance Weighting Factor / Significance Weighting / Case Amplification
    - Other Similarity Measures: Adjusted Cosine Similarity (Better Measure Compared to Pearson for Item-Based Recommender System) Spearman's Rank Correlation Coefficient, Mean Squared Difference
    - Neighborhood Selection: Efficiency & Accuracy Concern - Top K
        - MovieLens Dataset Example: [20, 50]
- Item-Based Nearest Neighbor Recommendation
    - Practical Concern (e.g. e-commerce sites): User-Based CF prediction computation in real time is impossible, while users' similarities could change dramatically due to relatively small number of overlapping ratings in practical scenarios
    - Core Ideology: Prediction for particular user's new item evaluation based on similar and already evaluated items
    - Standard Metric: Cousin Similarity Measure - Range [-1, +1]
        - Improved Version: Adjusted Cosine Measure - Take Difference in Average Rating Behavior into Account
    - Preprocessing-Based: Item Similarity Matrix - Offline Computation
    - Model-Based: Subsampling - Randomly Chosen Subset

- Ratings: Implicit & Explicit - Costs and Benefits
- Challenges
    - Data Sparsity: Each customer typically provides only a small fraction of items - Cold Start Problem: No enough information to compute and make prediction for unseen items or users
    - Proposed Solutions: Spreading Activation (Unstable Performance - Rating Matrix Density Related) / Default Voting (Damping Mechanism)
- Collaborative Recommendation Technique Categories
    - Memory-Based (i.e. User-Based): Rating Database held in memory for recommendation generation - Scalability Concern
    - Model-Based: Raw data offline preprocess (possibly with dimensionality reduction techniques) - Make prediction using precomputed model

Content-Based Recommendation - User Preference Match
- No need for large user community or rating history
- Content Definition - Description of Item Characteristics
    - Dice Coefficient - Keyword Similarity Measure
        - Weighted Item Characteristics Similarity Functions
    - Vector Space Model & TF-IDF - Irrelevant Information Removal
        - Stop words & stemming
        - Size cutoffs - Most Informative Words / Feature Selection
        - Phrases
    - Limitations - Missing Context & Misinterpretation Vectorization

Hybridization
- Input Data Requirement of Recommendation Algorithms

Paradigm	User profile and contextual parameters	Community data	Product features	Knowledge models
Collaborative	Yes	Yes	No	No
Content-based	Yes	No	Yes	No
Knowledge-based	Yes	No	Yes	Yes

- Designs - Monolithic & Parallelized & Pipelined
    - Monolithic: Incorporates Several Strategies within Single Implementation

    - Parallelized: Independent Operation - Separate Recommendation Lists - Combined (i.e. Weighted, Mixed, and Switching)
    - Pipelined: Cascade & Meta-level Hybridization

Recommender System Evaluation

- Validity & Reliability & Sensibility
    - Potential Issue: Practical Impact & Measurable Performance Improvement
- Methodology
    - N-Fold Cross-Validation
        - Leave One Out (i.e. Single Retrieval)
            - Tradeoff: Cost & Learning
        - All But N - Fixed Testing Set Size for Each User
        - Given N - Fixed Training Set Size
        - Given 0 - Empty Training Set & Testing Set Includes All Ratings (i.e. “Best Seller”)
- System Capability Evaluation
    - Prediction Task - Compute Missing Rating in User/Item Matrix
    - Classification Task - Ranked Lists of N Items Relevant to User
        - Typical Length Range [3, 10]
- Metrics
    - Prediction Accuracy
        - MAE
            - RMSE - Emphasize Larger Deviations
            - NMAE - Normalized MAE
    - Classification Accuracy
        - Precision & Recall
            - F1 Metric
        - Hit Rate
    - Rank Accuracy
        - Half-Life Utility Rank Score & Lift Index
    - Additional Metrics
        - User Coverage & Catalog Coverage
            - Intra-List Similarity