Hybrid Recommendation

Weike Pan

College of Computer Science and Software Engineering
Shenzhen University

Outline

Reference

- Chapter 5 of *Recommender Systems: An Introduction*
- Almost all the contents in this document are from the slides at http://www.recommenderbook.net/teaching-material/slides

What is Hybrid Recommendation

- *Hybrida* (Latin noun): denotes an object made by combining two different elements
- Combine several algorithmic implementations or recommendation components

Why Do We Need Hybrid Recommendation

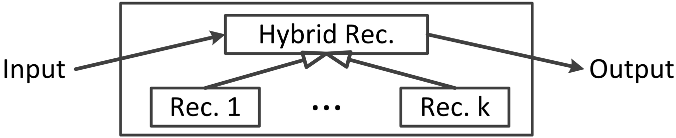
- Data sparsity problem
- Cold start problem
- Reach some desirable properties not (or only inconsistently) present in each single recommendation approach

Opportunities

- Netflix Prize competition
  - Hybridizing hundreds of different collaborative filtering algorithm techniques and approaches
- Robin Burke. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12:331-370, 2002.
  - It has 2399 citations shown in the results returned by Google's search engine on May 14, 2015.

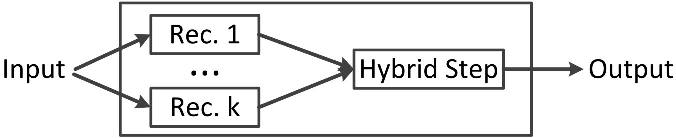
Monolithic Hybrids

- Incorporates aspects of several recommendation strategies in one algorithm implementation, e.g., factorization machine (FM) for ratings and content



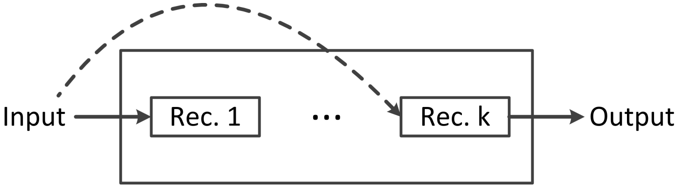
Parallelized Hybrids

- At least two separate recommender implementations
- Operate independently of one another and produce separate recommendation lists
- In a subsequent step, their output is combined into a final set of recommendations



Pipelined Hybrids

- At least two separate recommender implementations
- The output of one recommender becomes part of the input of the subsequent one
- Optionally, the subsequent recommender components may use parts of the original input data



Strategies

- Typically, data-specific preprocessing steps are used to transform the input data into a representation that can be exploited by a specific algorithm paradigm.
- Two specific strategies
  - Feature combination
  - Feature augmentation
- Stories of feature engineering of Douban (http://www.douban.com/)

Strategies

- Parallelized hybridization designs employ several recommenders side by side and employ a specific mechanism to aggregate their outputs.
- Three specific strategies
  - Mixed hybrids
  - Weighted hybrids
  - Switching hybrids
- Additional combination strategies for multiple recommendation lists such as majority voting may also be applicable.

Mixed Hybrids

- A mixed hybridization strategy combines the results of different recommender systems at the level of the user interface, in which results from different techniques are presented together.
- Therefore, the recommendation result for user u and item i of a mixed hybrid strategy is the set of tuples $(\hat{r}_{ui}, \kappa)$ for each of its k constituting recommenders, where $\kappa = 1, 2, \ldots, k$.

Weighted Hybrids (1/2)

- A weighted hybridization strategy combines the recommendations of two or more recommendation systems by computing weighted sums of their scores. Thus, given k different recommendation functions $\hat{r}_{ui}^{(\kappa)}$ with associated relative weights β_κ,

$$\hat{r}_{ui} = \sum_{\kappa=1}^{k} \beta_\kappa \hat{r}_{ui}^{(\kappa)} \tag{1}$$

  where item scores need to be restricted to the same range for all recommenders and $\sum_{\kappa=1}^{k} \beta_\kappa = 1$. Obviously, this technique is quite straightforward and is thus a popular strategy for combining the predictive power of different recommendation techniques in a weighted manner.

Weighted Hybrids (2/2)

- When applying a weighted strategy, one must ensure that the involved recommenders assign scores on comparable scales or apply a transformation function beforehand.
- In the extreme case, dynamic weight adjustment could be implemented as a switching hybrid. There, the weights of all but one dynamically selected recommenders are set to 0, and the output of a single remaining recommender is assigned the weight of 1.

Switching Hybrids

- Switching hybrids require an oracle that decides which recommender should be used in a specific situation, depending on the user profile and/or the quality of recommendation results.
- Even more adaptive switching criteria can be thought of that could even take contextual parameters such as users' intentions or expectations into consideration for algorithm selection.
- The quality of the switching mechanism is the most crucial aspect of this hybridization variant.

Strategies

- Pipelined hybrids implement a staged process in which several techniques sequentially build on each other before the final one produces recommendations for the user.
- The pipelined hybrid variants differentiate themselves mainly according to the type of output they produce for the next stage. In other words, a preceding component may either preprocess input data to build a model (e.g., domain adaptation methods) that is exploited by the subsequent stage or deliver a recommendation list (e.g., re-ranking methods) for further refinement.
- Two specific strategies
  - Cascade hybrids
  - Meta-level hybrids

Cascade Hybrids (1/2)

- Cascade hybrids are based on a sequenced order of techniques, in which each succeeding recommender only refines the recommendations of its predecessor.
- However, they may not introduce new items - items that have already been excluded by one of the higher-priority techniques - to the recommendation list. Thus, cascading strategies do have the unfavorable property of potentially reducing the size of the recommendation set as each additional technique is applied.
- As a consequence, situations can arise in which cascade algorithms do not deliver the required number of propositions, thus decreasing the system's usefulness.
- Therefore, cascade hybrids may be combined with a switching strategy to handle the case in which the cascade strategy does not produce enough recommendations.

Cascade Hybrids (2/2)

- Stories about search engines ...

Meta-level hybrids

- In a meta-level hybridization design, one recommender builds a model that is exploited by the principal recommender to make recommendations.
- The Fab system (Balabanovic and Shoham 1997) employs a content-based recommender that builds user models based on a vector of term categories and the users' degrees of interest in them. The recommendation step, however, does not propose items that are similar to the user model, but employs a collaborative technique. The latter determines the user's nearest neighbors based on content models and recommends items that similar peers have liked.
- Stories about Tencent's user profiling ...

Summary (1/3)

- Monolithic hybridization design
  - Feature combination hybrids
  - Feature augmentation hybrids
- Parallelized hybridization design
  - Mixed hybrids
  - Weighted hybrids
  - Switching hybrids
- Pipelined hybridization design
  - Cascade hybrids
  - Meta-level hybrids

Summary (2/3)

- No single hybridization variant is applicable in all circumstances, but it is well accepted that all base algorithms can be improved by being hybridized with other techniques.
- For instance, in the Netflix Prize competition, the winners employed a weighted hybridization strategy in which weights were determined by regression analysis (Bell et al. 2007).
- Furthermore, they adapted the weights based on particular user and item features, such as number of rated items, that can be classified as a switching hybrid that changes between different weighted hybrids according to the presented taxonomy of hybridization variants.

Summary (3/3)

- Monolithic designs are advantageous if little additional knowledge is available for inclusion on the feature level. They typically require only some additional preprocessing steps or minor modifications in the principal algorithm and its data structures.
- Parallelized designs are the least invasive to existing implementations, as they act as an additional post-processing step. Nevertheless, they add some additional runtime complexity and require careful matching of the recommendation scores computed by the different parallelized algorithms.
- Pipelined designs are the most ambitious hybridization designs, because they require deeper insight into algorithm's functioning to ensure efficient run-time computations. However, they typically perform well when two antithetic recommendation paradigms, such as collaborative and knowledge-based, are combined.

Homework

- Read chapter 5 of *Recommender Systems: An Introduction*
- Read the UMUAI 2002 paper by Robin Burke.
  - Robin Burke. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12:331-370, 2002.