FINTECH FORUM

Exploring your future @HKUST

Hi stranger! I know your taste before you tell!

-- Applying transfer learning to improve article recommender system performance

Authors and Contacts

CHEN Yuxiang
ycheneu@connect.ust.hk

LIN He
hlinam@connect.ust.hk

XU Mengdi
md.xu@connect.ust.hk

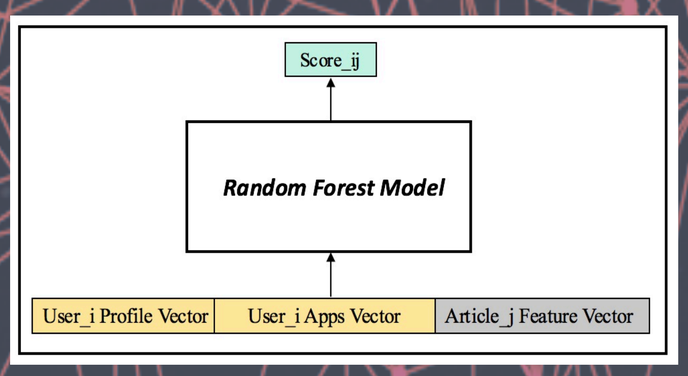
ZHENG Dingyi
dzhengaf@connect.ust.hk

Problem Description

"Without knowing anything about someone's reading history, how would you know their tastes of readings? "
This is a dual cold start problem - recommending fresh articles that have not yet been read by anyone, to new users who have not yet read any articles.
It is hard to get an accurate recommendation by traditional recommendation systems due to lack of preference information in the article domain.
However, the user profile data (e.g. gender, age, etc.) as well as their mobile application installation history can be informative.
We are able to analyze user's preference of articles base on their behavior in downloading different types of mobile applications.
Therefore, we can recommend articles to each user even without knowing their reading history.

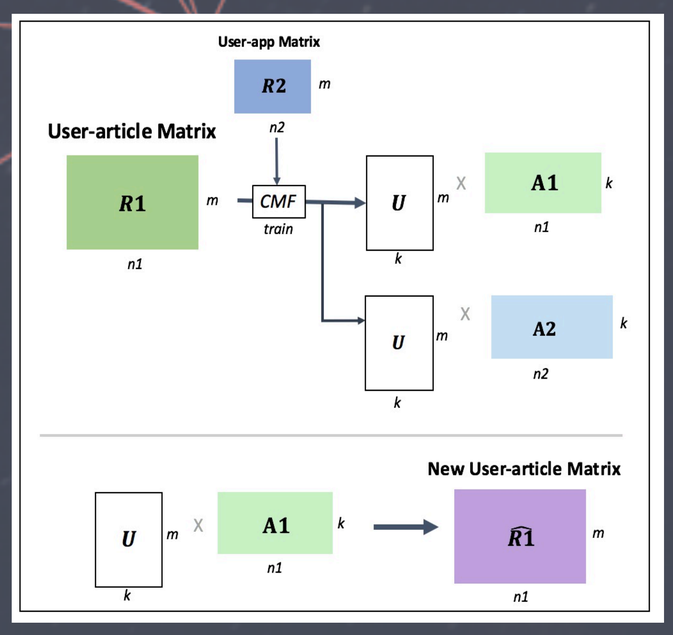
Solution 1: Supervised Learning

Our baseline uses a random forest to predict a user's reading time on each article. Only five articles with the longest predicted reading time will be recommended to the corresponding user.



The model performance has precision 0.167% and recall 0.058% for top 5 recommendations.

Solution 2: Pattern Transfer Learning



This model applies collective matrix factorization on both rating matrices simultaneously to transfer user's rating patterns.

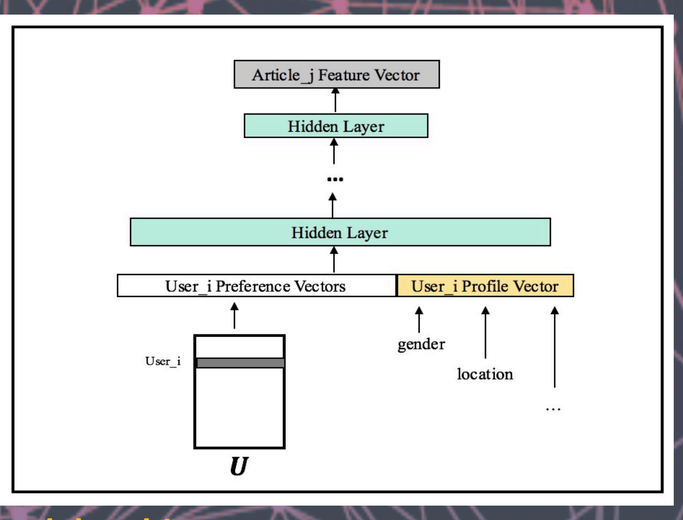
$$\text{loss} = \frac{1}{2}\|J \circ (R_1 - UA_1^T)\|^2 + \frac{\alpha}{2}\|R_2 - UA_2^T\|^2 + \frac{\beta}{2}(\|U\|^2 + \|A_1\|^2 + \|A_2\|^2)$$

Training method:
By minimizing the regularized squared error on the set of known ratings, it generates 3 feature matrices characterizing apps, articles and users by k-dimensional vectors of factors.

Prediction:
we calculate dot product to obtain a denser user-article rating matrix. The final product represents users' preferences on each article.

Performance:
The model has 0.374% in precision and 0.129% in recall, which is better than the baseline.

Solution 3: Domain Transfer Learning



Model architecture:
By training a deep neural network, we can find a bridge from user-app domain to user-article domain.

Training input:
user profile data and embedded user preference vector

Training target:
embedded vector of their favorite article

Testing input:
user profile data and averaged embedded user preference vectors of KNN.

Testing model output:
guess for an embedded vector of the user's favorite article

Testing output:
nearest 5 new articles to the guess.

Solution 4: Enhancing with Deep Learning on Scaling Issues

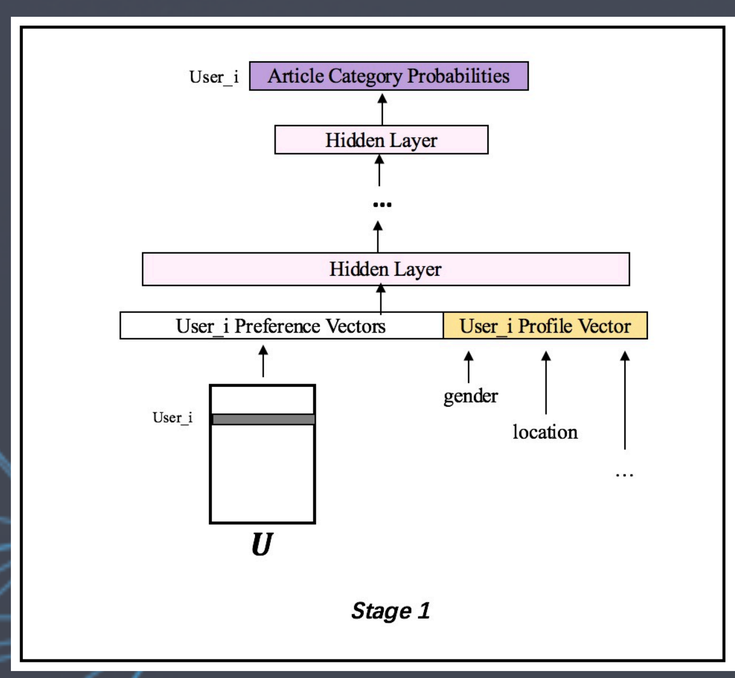
Model interpretation:
we explored on splitting the recommendation system into two stages due to large scale and freshness of the article data. In stage 1, we detailed a deep candidate selection model; then we generated a deep ranking model in stage2.

Stage 1

Input: embedded user preference vectors and user profile feature vectors
Output: probabilities the user would like to read in each category.
Further result: generate a small group of article candidates for each user by preferred article category for stage2.

Stage 2

Input: article candidate embedded feature vector and corresponding article title info and user profile vector
Output: predicted score representing the reading time
Further result: recommended articles of top-5 scores to the corresponding user.



Prediction Approach for Testing Data

	KNN users:	KNN articles:
Idea:	find K nearest neighboring users from training set for each user in testing set	find K nearest neighboring articles from training set for each article in testing set
Features for measuring distances:	**Categorical:** age, gender, city, language, mobile's brand, etc. (num=7) **Numeric:** number of downloaded for each app category. (num=20)	**Categorical:** publisher (num=1) **Numeric:** l1 category weights, l2 category weights (num=193)

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

HKUST-MSRA Joint Lab

FDT

Microsoft Research