Learning Transfer

Title: 0 Deep transfer learning with multimodal embedding to tackle cold-start and sparsity issues in recommendation system

Summary:
- Method proposed, offline deep learning, second online phase
- General Problems, sparsity, cold start, generalizability
- Classic Techniques, Collaborative Filtering, Content-based, Hybrids?
- Cold start solutions exist
- DNNs improve performance
- Metadata can make user dense
- When new items are added RS faces difficulty making inferences
- Purposes Deep Transfer Learning with Multimodal Embedding
- Model presented: Deep Transfer Learning, Multimodal embedding, offline, recommender model online

Terms:
- RS - Recommender Systems
- Cold-start
- Sparsity
- DLRS – deep learning-based recommender systems

Network Architecture in Paper

Title: 1 Classification and Comparison of the Hybrid Collaborative Filtering Systems (From 0)

Summary:
- Overview of use of CF with other methods know as a hybrid
- Hybrid types, UB CF, IB CF, Demographic CF, Knowledge Based CF, Semantic Based CF
- CF and CB in order to cope with new item, sparsity and gray sheep problems of CF
- CF with DM recommendations can reduce sparsity problem of CF
- Combining CF with KB: It will avoid the disadvantages of a CF recommender, such as such as new user/item problem, sparsity, gray sheep, the requirement to be initialized with a large database of users' ratings, and the possibility to generate invalid recommendations when a user's interests change. However, this system requires knowledge engineering with all of its attendant difficulties
- Combining CF with SB: Combining semantic similarity and rating similarity between items provides two primary advantages over pure CF. First, the semantic attributes for items allows the system to make inferences based on the underlying reasons for which a user may or may not be interested in a particular item. Secondly, in the case of new item or in very sparse data sets, the system can still use the semantic information to provide reasonable recommendations for users
- Combining CF with CA: CA systema can make recommendations that best match with users' preferences and needs at the right moment and in the right place. Adomavicius and Tuzhilin [83]

presented three different approaches for incorporating contextual information into recommendation process: pre-filtering, post-filtering and contextual modeling

Terms:

- CF – Collaborative Filtering
- Active user – one who uses RS
- Gray Sheep – do not agree with any group
- Scalability – ability to grow
- UB – User based CF
- IB – Item based CF
- Demographic CF
- Knowledge Based CF
- Semantic Based CF
- Context Aware CF

Review Table in Paper

Title: 2 Cold-Start Item and User Recommendation with Decoupled Completion and Transduction (From 0)

Summary:

Terms:

Title: 3 DDTCDR: Deep Dual Transfer Cross Domain Recommendation

Summary:

To address these problems, researchers propose to use cross domain recommendation through transfer learning approaches that learn user preferences in the source domain and transfer them to the target domain. For instance, if a user watches a certain movie, we will recommend the original novel on which the movie is based to that user.

Specifically, we propose Deep Dual Transfer Cross Domain Recommendation (DDTCDR) model that learns latent orthogonal mappings across domains and provides cross domain recommendations leveraging user preferences from all domains.

As shown in Table 4, 5 and 6, the proposed DDTCDR model significantly and consistently outperforms all other baselines in terms of RMSE, MAE, Precision and Recall metrics across all the three domain pairs.

As an important tool to provide recommendations, matrix factorization methods associate user-item pairs with a shared latent space, and use the latent feature vectors to represent users and items. In order to model latent interactions between users and items as well as feature information, it is natural to generalize matrix factorization methods using latent embedding approaches. However, in the cross domain recommendation application, we may not achieve the optimal performance by uniformly applying neural network model to all domains of users and items due to the cold-start and sparsity problems.

To address these problems and improve performance of recommendations, we propose to combine dual transfer learning mechanism with cross domain recommendations, where we transfer user preferences between the source domain and the target domain simultaneously.

Terms:

Network architecture in paper.

Recommender

Title: 0 Optimizing Recommender System: Literature Review

Summary:

- Types
  - Personalized Recommendation System This type of recommendation system makes use of user profile and some contextual parameters of users and provide personalized recommendations to users
  - Collaborative Recommendation System This type of recommendation system makes use of the user profile, some contextual parameters, and data of the community to which the user belongs. It recommends a similar product to a user which other users of his community are buying.
  - Content-based Recommendation System This type of recommendation system makes use of user profile, contextual parameters as well as features of the product. Based on this it recommends the product to the user which has the same feature as the product he has already purchased before
- Knowledge-based Recommendation System This type of recommendation system makes use of user profile, contextual parameters, product features and knowledge models which keeps track of certain event in user's demographics and accordingly do the recommendations. For example, in the case of the birthday of the user, it recommends a certain product. In the case of festivals related to the user's religion like Diwali, it recommends a certain product to the user.
- Hybrid Recommendation This type of recommendation system makes use of a combination of all four recommendation system methods/parameter and try to recommend the best suitable product for the user
- Problems
  - Cold-start
  - Data sparsity
  - Scalability This characteristic is difficult to achieve. It consists of several users and products. Nowadays many eCommerce systems consist of millions of customers. It is difficult to recommend them correct sets of products from their hundreds of products. Improving scalability was addressed in some studies.
- Diversity Our system must understand diversity. Our system should have a recommendation list that consists of all products which are similar to each other. If one customer is not interested in any of the products in that list, he might not be interested in any of them and gets no value from that recommendation list.
- Recommendation Techniques

N2VSCDNNR
Collaborative Filtering
Collaborative filtering using ontology and dimensionality reduction techniques
Structural Balance Theory-based E-commerce Recommendation over Big Rating Data
Adaptive Deep Modelling of Users and Items Using Side Information for Recommendation
Future Research
We need to adapt the latest clustering algorithm to increase efficiency. We need to explore more networking algorithms for data sparsity. The model should be more robust and scalable. The threshold and another parameter of the model should be minimum and dynamic. Time complexity is still high in most of the models. Computational complexity is still high in most of the models.

Terms:
Diversity: Variability of users
N2VSCDNNR:

Summary of Research Methods Table

Title: 1 A systematic literature review of multicriteria recommender systems

Summary
Foundational papers: RSs were initially conceived at the beginning of the 90s (Goldberg et al. 1992)
The most widespread categories of RSs are content-based, collaborative fltering, knowledge-based, and hybrid
Review of the number of studies of each problem

Combining Parallel Networks

Title: 0 Parallel Recurrent Neural Network Architectures for Feature-rich Session-based Recommendations

Summary:
In traditional recommender systems algorithms it is often assumed that user history logs (e.g. clicks, purchases or views) are available. We show that this assumption does not hold in many real-world recommendation use cases
In this section we describe the proposed parallel RNN (pRNN) architectures that utilize item representations (features) for session modeling. A p-RNN consists of multiple RNNs, one for each representation/aspect of the item (e.g. one for ID, one for image and one for text). The hidden states of these networks are merged to produce the score for all items. We also introduce baseline

architectures, naive approaches for using the different item representations.
Baseline architecture: ID only, Feature only, Concatenated input, p-RNN architecture, Parallel shared-W, Parallel interaction
Methods of Training, Simultaneous, Alternating, Residual, Interleaving

Terms:
p-RNN: Parallel Residual Neural Networks
Interleaving: Alternating training in each minibatch

Network Architecture Given

Training

Title: 0 Learning to Recommend from Sparse Data via Generative User Feedback

Summary:
Traditional collaborative filtering (CF) based recommender systems tend to perform poorly when the user-item interactions/ratings are highly scarce. To address this, we propose a learning framework that improves collaborative filtering with a synthetic feedback loop (CF-SFL) to simulate the user feedback
The proposed CF-SFL framework can be interpreted as an inverse reinforcement learning (IRL) set-up, in which the recommender learns to recommend the user items (policy) with the estimated guidance (feedback) from the proposed virtual user
To overcome the aforementioned problems, we propose a collaborative filtering framework with a synthetic feedback loop (CF-SFL).
The feedback generator connects the reward estimator with the recommender module via generating a feedback embedding,
The collaboration between the recommender policy $\pi\theta$ and the feedback generator $F\psi$ towards a better predictive recommender. The adversarial game between the recommender policy $\pi\theta$ and the reward estimator $R\phi$.
Evaluation Metrics We employ Recall@r2 together with NDCG@r3 as the evaluation metric for recommendation, which measures the similarity between the recommended items and the ground truth. Recall@r considers top-r recommended items equally, while NDCG@r ranks the top-r items and emphasizes the importance of the items

Terms:
Recall@r2: recommendation engine
NDCG@r3: Evaluation engine

Datasets:
Netflix-Prize (Netflix), another user-movie ratings dataset collected by the Netflix Prize

Network Architecture Given

Title: 1 LEARNING CONSISTENT DEEP GENERATIVE MODELS FROM SPARSE DATA VIA PREDICTION CONSTRAINTS

Summary:

- Our framework optimizes model parameters to maximize a variational lower bound on the likelihood of observed data, subject to a task-specific prediction constraint that prevents model misspecification from leading to inaccurate predictions. We further enforce a consistency constraint, derived naturally from the generative model, that requires predictions on reconstructed data to match those on the original data.
- Our approach is inspired by the prediction-constrained framework recently proposed for learning supervised topic models of "bag of words" count data

Terms:

- Autoencoder:
- VAE: Variational Auto-Encoder
- GAN: Generative Adversarial Network

Data Mining

Title: 0 APPLICATION OF BIG DATA IN EDUCATION DATA MINING AND LEARNING ANALYTICS – A LITERATURE REVIEW

Summary:

- Entry of open source projects in mobile computing has led to low cost smartphones
- The traditional RDBMS tools will be unable to store or process such Big Data. To overcome this challenge, databases that don't use traditional SQL based queries are used. Compression technology is used to compress the data at rest and in memory.
- Standard Big Data tools used, Regression, Nearest Neighbor, Clustering, Classification, MongoDB, Hadoop, MapReduce, Orange, and Weka.

Terms:

- Skill Estimation
- Attrition Risk
- Behavior Detection
- Game-based Learning
- Student Modeling

Title: 1 Big Educational Data & Analytics: Survey, Architecture and Challenges

Summary:

- Classification: Table of Research
- Platforms: LMS, MOOC, learning object repository (LOR), OCW, OER, social media, linked data and mobile learning
- Databases: Hadoop, Spark and Samza
- Approaches: s collaborative filtering, content-based filtering and hybrid recommendation systems

Terms:

- LMS – Learning Management Systems
- MOOC

Neural Networks

Title: 0 Deep Collaborative Filtering Based on Outer Product

Summary:

Terms:

Title: 1 Relational Learning via Collective Matrix Factorization

Summary:

- Many relational domains involve only one or two entity types: documents and words; users and items; or academic papers where links between entities represent counts, ratings, or citations.
- In domains with more than one relation matrix, one could fit each relation separately; however, this approach would not take advantage of any correlations between relations. For example, a domain with users, movies, and genres might have two relations: an integer matrix representing users' ratings of movies on a scale of 1–5, and a binary matrix representing the genres each movie belongs to. If users tend to rate dramas higher than comedies, we would like to exploit this correlation to improve prediction.
- Factorization algorithm can be defined by choices in a set of algorithms

Terms:

Title: 2 Personalized Recommendation via Cross-Domain Triadic Factorization

Summary:

- Slices of domain-specific matrices with heterogeneous items are transformed into a cubical tensor containing virtual items with identical length.

All domains are horizontally concatenated.

We need to reconstruct all missing values for prediction but the standard fitting algorithm for PARAFAC2 is based on the complete data. Therefore, we need to design a new fitting algorithm which allows dealing with missing data. Thus, we apply an Expectation Maximization (EM) sub-procedure into the fitting algorithm to handle the incomplete data by iteratively imputation after each full cycle of updates.

Here, we employ the genetic algorithm (GA) [3] to find such optimal weights assignment. We fix the weight on target domain to be 1 so various weights assignment on auxiliary domains act as the individuals in the population

Terms:

Cross-Domain Collaborative Filtering

Title: 3 Cross-Domain Collaborative Filtering with Factorization Machines

Summary:

Cross-domain Collaborative Filtering (CDCF) methods exploit knowledge from auxiliary domains (e.g., movies) containing additional user preference data to improve recommendation on a target domain (e.g. books). While relying on a broad scope of existing data in many cases is a key to relieving the problems of sparse user-item data in the target domain, CDCF can also simultaneously benefits different data owners by improving quality of service in different domains

– MF-SGD (D): Matrix Factorization method using SGD learning algorithm on single domain D. – FM-X (D): Factorization Machine method on single domain D based on learning algorithm X (SGD, ALS or MCMC). – FM-All-X (D): Combining all rating data into single domain (blind combination) and testing target domain D by using FM with algorithm X. This approach simply increases the size of training data by including the rating data of all domains. In other words, the feature vector x is represented as in equation (2) and all items in different domains are treated the same. – FM-X (DT , {DA}): Factorization Machine method on target domain DT and auxiliary domains {DA} based on algorithm X. – PF2-CDCF: The Cross-Domain CF method which is proposed by Hu et al. [2] on the same dataset

Terms: