Final Project Proposal

Predict decisions on Weibo with Factorization Machines

Kyuhyun Cho(20133677) Seunghwan Baek(20133331)

Joongyoun Lee(20115494) Hyeonjun Jo(20133693)

1. PROBLEM

How can we get to know if a person chooses something or not? Basically, we can ask them directly. However, how about guessing the decision without explicit response? It is a different problem. We may need to gather data which seems to be related on the decision. At this point, we have thought that A.I can do the 'guessing' task better than human. They can collect related information, look into it, and compile statistics on it. Then, they may find some rules and finally make a correct decision. This is what we are going to do in this project. There are amazingly countless data on social networks, especially Weibo. We are going to create an A.I which predicts user's decision whether he/she may interests on recommended item (a person, a group or a product in Weibo). The predictor may perform a regression based on training data, but there could be a lot of noise in the raw data. Thus, we have to choose subset from the entire dataset to train A.I. We believe that it should be the most challenging problem to correctly select data. The second problem we have to consider is the "overfitting". If our predictor is too optimized on training data, a precision on the test data could be very low. We have to deal with this problem using regularization or something like that. The third most important problem of this proposal is the "cold-start" problem. It happens due to the lack of information for a large number of users in testing dataset.

2. DATA

We are going to pick sub-data among entire dataset although we do not decide yet. As we already mentioned, there should be lots of noise in the raw data. Thus, we will perform a preprocessing on selected dataset to remove noise. Some dummy users or duplicate data could be removed at this stage. We are also concerning the dimensionality reduction technique such as PCA not only to enhance precision, but also to reduce prediction and training time.

3. METHOD

We will adopt Factorization Machines which are a new model that combines advantages of SVM with factorization models because of following advantages of FM [1] :

1. FMs allow parameter estimation under very spare data.
2. FMs have linear complexity and can scale to large datasets with 100M of training instances.
3. FMs can mimic state-of-the-art models like biased MF, SVD++, PITF or FPMC because FMs are a general predictor that can work with any real valued feature vector.

Those merits make it acceptable to use FMs to deal with Weibo dataset which should make sparse and very large matrix. Despite of those advantages of FMs, we believe that FM itself can not show adequate precision due to the complicated features in the Weibo data. Thus, we are going to consider some features to enhance our basic predictor. The first one is temporal dynamics. This means trends of items. If an item is recommended to a user and the item is very trendy at that time, the probability of user's acceptance should be high. The second is a users' active period. It is another aspect of temporal information on the Weibo data. If a user is very active at certain period of time, then he/she may accept item which is recommended at that period of time with high probability. The last is the implicit feedback. Training data may generate very sparse matrix due to the very limited explicit information. It should degrade our predictor in terms of precision. Thus, we are going to focus more on users who are similar to the target user. If there are some common items among the users, we could guess that the target user also should be interested in the items.

4. IMPLEMENTATION

We are going to employ C++ as a primal programming language, although we may also use Python partially to deal with Python friendly tasks. The main machine learning library is a libFM which is an implementation of FM, while other ML libraries such as mlpack or MLC++ may be adopted for supplementary purpose.

5. REFERENCES

[1] Steffen Rendle, "Factorization Machines," in the Institute of Scientific and Industrial Research Osaka University, Japan

[2] Steeffen Rendle, "Factorization Machines with libFM," in University of Konstanz