Optimizing the Feature Selection Process for Better Accuracy in Datasets with a Large Number of Features (Student Abstract)

Xi Chen,[1] **Afsaneh Doryab**[2]
[1]Carnegie Mellon University, [2]University of Virginia
xc3@andrew.cmu.edu, ad4ks@virginia.edu

Abstract

Most feature selection methods only perform well on datasets with relatively small set of features. In the case of large feature sets and small number of data points, almost none of the existing feature selection methods help in achieving high accuracy. This paper proposes a novel approach to optimize the feature selection process through Frequent Pattern Growth algorithm to find sets of features that appear frequently among the top features selected by the main feature selection methods. Our experimental evaluation on two datasets containing a small and very large number of features shows that our approach significantly improves the accuracy results of the dataset with a very large number of features.

Introduction

Feature selection methods often improve the performance of the predictors, help select the most cost-effective features, and create a better understanding of the underlying process that generated the data. However, none of these methods can provide a stable set of features when the number of features is extremely large compared the number of data instances. In this paper, we propose a method that selects the most stable sets of features with relatively great predictive performance by utilizing the FP-Growth algorithm (Han, Pei, and Yin 2000). Frequent patterns are itemsets that appear in a data set with frequency no less than a user-specified threshold (Han et al. 2007). For example, a set of items, such as milk and bread, that appear frequently together in a transaction dataset, is a frequent itemset. In our case, we aim to mine sets of features that frequently appear together in the transaction dataset containing the top ranked features selected by different feature selection methods. The following sections describe our approach in details.

FS-FPG Algorithm

We first apply a set of feature selection methods on the dataset within the Cross-Validation process. We divide the original dataset into k folds and for each fold, we apply each feature selection method on the training set to obtain a set of features. Then the feature sets generated are transformed into a transaction dataset to which we apply FP-Growth algorithm to generate rules and frequent sets of features. In order to select useful feature sets from the set of all possible ones, we adopted minimum thresholds on support and confidence as constraints on measures of significance. Support and Confidence are defined as follows. Let A and B be two sets of items. An association $(A \rightarrow B)$ exists if items in A and B frequently appear together in transactions. Support is the percentage of transactions that contain both A and B, whereas confidence is the percentage of transactions containing A that also contain B (Han, Pei, and Yin 2000), that is, support $(A \rightarrow B) = P(A \cup B)$ and confidence $(A \rightarrow B) = P(B|A)$. Those sets of features satisfying the minimum support and confidence threshold are stored for testing at the end.

procedure ITEMSETS-GENERATING(D, min_sup, sc_thres)
divide D into k folds
$freq\_Itemsets \leftarrow []$
$feature\_sel\_methods \leftarrow [\text{fs}_1, fs_2, ...fs_n]$
for $i \leftarrow 1$ **to** k
  for each fs_j *in feature_sel_methods*
  do *Apply* fs_j *on the ith training set*
  do *Get the scores of the features*
  do *Normalize the scores of the features in a 0 to 1 scale*
  do $Items \leftarrow$ [*features with scores more than sc_thres*]
  do $freq\_Itemsets.append(Items)$
$DB \leftarrow$ *Transaction Database constructed from freq_Itemsets*
Call FP-growth(DB, min_sup)

Experiment

Datasets

In this work, we used two datasets for our experiment. The first dataset (D1) contains smartphone and Fitbit data from a cohort of 140 students. The dataset includes a large number of features (7731 in total) and only 140 instances with each instance containing data from a student. The ratio of number of features to number of instances is so big that feature selection becomes essential and indispensable in creating a good model for further prediction. The response variable of this dataset is post-loneliness which can be interpreted according to the UCLA loneliness scale. For the sake of com-

FS Methods	XGB	KNN	SVM	NN	LR
Chi-square	0.37	0.56	0.29	0.47	0.19
MI	**0.68**	**0.64**	0.31	0.39	0.38
F-test	0.57	0.42	0.30	0.40	0.19
RF	0.63	0.52	0.27	0.54	0.34
RLR	0.49	0.50	0.52	0.49	0.21

Table 1: Cross-Validated accuracy of each feature selection method for D1

FS Method	XGB	KNN	SVM	NN	LR
Chi-square	0.87	0.79	0.83	0.87	0.85
MI	**0.97**	0.94	0.92	0.94	0.92
F-test	0.94	0.92	0.58	0.90	0.90
RF	**0.95**	0.93	0.69	0.74	0.90
RLR	0.91	0.90	0.75	0.85	0.87

Table 2: Cross-Validated accuracy of each feature selection method for D2

parison, we used another dataset (D2) – The ExtraSensory Dataset [1], which contains only 225 features but 3880 instances. This dataset is for behavioral context recognition of users who were engaged in their regular natural behavior from mobile sensors. The response variable of this dataset is sitting, which is binary and indicates whether the user is sitting or not, as detected by the sensors. We pre-processed and cleaned the datasets and dropped features with more than 30% missing values. We then filled the rest of missing values by the averages of the corresponding columns to keep the data in reasonable range.

Procedure

We applied five widely used feature selection methods on our datasets to select the top features by each algorithm. First, we set k to be 10 and divide the dataset into 10 folds and applied feature selection methods, including Chi-square, Mutual Information, Randomized Logistic Regression, Anova, and feature selection by tree structure(e.g: feature importances provided by the random forest (RF)) to each fold. To reduce the impact of different score ranges we normalized the scores given by each feature selection method to be in the range of 0 to 1. We set sc_thre to 0.5 to obtain the highest ranked ones. We generated new datasets with the selected features and applied several representative classification algorithms including Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Neural Network (NN), Logistic Regression (LR) and Xgboosting (XGB) to each transformed dataset and measured the accuracy using 10-fold cross-validation. In the next step, we recorded the sets of features provided by each feature selection method during each fold which generated 50 sets in total. We then transformed these sets into a transaction dataset and applied the FP-growth algorithm on it to mine association rules and frequent itemsets for the features of the dataset. We recorded the maximum support count among all the results and kept the frequent itemsets with support counts more than or equal to the maximum support count divided by 2 for further testing.

[1] link to the dataset: http://extrasensory.ucsd.edu/

Dataset	Feature Sets	SVM	KNN	XGB	NN	RF	LR
D1	[f1,f2,f3]	0.68	0.61	**0.71**	0.68	0.60	0.69
D1	[f4,f1,f3]	0.70	0.60	0.68	0.68	0.66	0.68
D1	[f4,f1]	**0.73**	0.69	0.65	0.68	0.65	0.68
D1	[f4]	0.70	0.68	0.62	0.68	0.57	0.68
D1	[f4,f5,f6]	0.70	0.68	0.60	0.67	0.63	0.66
D1	[f6]	0.69	0.70	0.65	0.68	0.58	0.48
D1	[f1,f5, f6]	0.70	0.68	0.60	0.68	0.63	0.62
D2	[f1,f2,f3]	0.68	**0.70**	0.63	0.68	0.63	0.68
D2	[f1,f4]	0.68	0.66	0.65	0.67	0.65	0.68
D2	[f5,f3]	**0.69**	**0.69**	0.61	0.68	0.57	0.68
D2	[f6,f5,f4, f2]	0.68	0.68	0.66	0.67	0.65	0.67
D2	[f5,f4,f3]	0.68	0.68	0.65	0.68	0.63	0.67

Table 3: Cross-Validated accuracy of each frequent feature set for D1 and D2 after applying FS-FPG

Results

The 10-fold cross-validated accuracy results after the common feature selection methods but without applying the FS-FPG algorithm are presented in Tables 1 and 2. The accuracy results for the D1 dataset with a large number of features is low for most classification methods 1 while those results are generally high for D2 with a small set of features 2. After applying FS-FPG algorithm, we notice opposite results (Table 3), i.e., the accuracy of classification for the D1 dataset increases significantly for the majority of the tested classifiers but the accuracy is decreased in the D2 results.

Conclusion

Our results show that our approach in reducing the number of features using the FP-Growth algorithm improves the performance in datasets with a large number of features and small set of instances while it does not have the same effect when the number of features is relatively small. We plan to optimize the algorithm and replicate the results with more datasets.

References

Han, J.; Cheng, H.; Xin, D.; and Yan, X. 2007. Frequent pattern mining: current status and future directions. *Data mining and knowledge discovery* 15(1):55–86.

Han, J.; Pei, J.; and Yin, Y. 2000. Mining frequent patterns without candidate generation. In *ACM sigmod record*, volume 29, 1–12. ACM.