Multi-Stage Ensemble and Feature Engineering for MOOC Dropout Prediction

Jeong-Yoon Lee
Conversion Logic
jeong@conversionlogic.com

Andreas Toescher
Opera Solutions
andreas.toescher@commendo.at

Kohei Ozaki
Recruite Technologies
kohei_ozaki@r.recruit.co.jp

Mert Bay
Conversion Logic
mert@conversionlogic.com

Michael Jahrer
Opera Solutions
michael.jahrer@commendo.at

Tam T. Nguyen
Institute for Infocomm Research
nguyentt@i2r.a-star.edu.sg

ABSTRACT

In this paper, we present the winning solution of KDD Cup 2015, where participants are asked to predict dropouts in a Massive Open Online Course (MOOC) platform. Our approach demonstrates best practices in feature engineering dealing with complex real world data, and pushes forward the state-of-the-art ensemble technique. We begin with feature engineering and extracted XXX and YYY features from raw student activity logs, course enrollment, and course material data. Then, we train 64 classifiers with 7 different algorithms and different subsets of extracted features. Lastly, we blend predictions of classifiers with the multi-stage ensemble framework. Our final solution achieves AUC scores of 0.90918 and 0.90744 on the public and private leaderboards respectively, and puts us to the 1st place out of 821 teams.

CCS Concepts

•**Computing methodologies** → **Supervised learning by classification; Ensemble methods;** *Cross-validation;*

Keywords

KDD Cup, Feature Engineering, Ensemble Learning

1. INTRODUCTION

Since 1997, KDD Cup has been one of the most prestigious competitions in knowledge discovery and data mining, where experts around the world from both industry and academia compete with each other with best modeling practices to solve real world challenges in complex data sets.

Massive Open Online Course (MOOC) platforms aim at providing the mass population with open access to quality education. Although their initial success in a few courses, MOOC platforms have struggled with extremely high dropout rates. Perna et al. reported that the average completion rate is 4% among 1 million students across 16 Coursera courses offered by the University of Pennsylvania from June 2012 to June 2013 [2]. If we identify those who are likely to drop out, we can engage with and help them complete courses successfully.

Therefore, KDD Cup 2015 asks participants to predict the likelihood of dropout for students based on their activity logs, course enrollment, and course material data provided by XuetangX, one of the largest MOOC platforms in China. Activity logs of 200,906 enrollments from 112,448 students across 39 courses are provided. Each activity is described by 6 fields of the username, course ID, timestamp, source, event, and object. For each object, 3 additional fields of the category, children, and start date are provided. The training set consists of 8,157,278 logs from 120,543 enrollments with the target variable indicating if a student dropped out. The test set consists of 5,387,848 logs from 80,363 enrollments. The full description of the data sets is available in [1].

Our final solution is a joint work from 9 data scientists, distributed around the world. The pipeline from raw data to final solution is as follows:

- Hand crafted feature engineering (most of hard work)
- Automatic feature design (autoencoder)
- Individual models (gbm, nn, factor model,..)
- Stage-I ensemble (blends individual models)
- Stage-II ensemble (blends stage-I ensemble models)
- Stage-III ensemble (blends stage-II ensemble models)

The rest of the paper is organized as follows. Section 2 describes our feature engineering approach. Section 3 introduces various classification algorithms used to train single classifiers. Section 4 overviews our multi-stage ensemble framework. Section 5 shows our final solution. Section 6 concludes the paper.

2. FEATURE ENGINEERING

Our team members extracted 7 feature sets, F1 F7 from raw data independently.

2.1 Common Features

There are common features across 7 feature sets as follows:

- blah

KDD '16 August 13–17, 2016, San Fransisco, CA, USA
© 2016 ACM. ISBN 978-1-4503-2138-9…$15.00
DOI: 10.1145/1235

2.2 F1

Features generated by Song and Kohei can be classified as follows:

- Enrollment-based features (No.1-8)
- Username-based features (No.9-18)
- Username-based features for each courses (No.19-25)
- Features based on 10 days after the end date of course (No.26-35)
- Features based on 1 day after the end date of a course (No.36-45)
- Day-level features (No.46)
- Day-level features using target variables (No.47-58)

Full list of features generated by Song and Kohei are described in Table 1. (just listing them for now. TBD in detail).

2.3 F2

Peng and Xiaocong features are comprised of the following parts:

- Visit time(hour, day) set features (including time span and max absent days)
- Act(event, object) counting features (some uses missed content counts)
- Course drop rate
- Number of courses the user enrolled
- Minimum time interval between time points(first visit, last visit, course begin, course end, 10 days after course end) of current course and another enrolled course
- Active days between course end and 10 days after course end
- Active days between last visit and course end
- Number of courses ended after current course end

The full feature list could be found in Table 2.

2.4 F3

Tam feature.

2.5 F4

Features generated by Michael Jahrer are in sparse format:

- uID (0-112,447)
- cID (112,448-112,486)
- uIDcnt (112,487-112,487)
- eIDcnt (112,488-112,488)
- eID → sID (112,489-112,490)
- eID → evID (11,2491-112,497)
- eID → oIDCnt (112,498-139,443)
- eID → tIDCnt (139,444-139,635)
- uID: floor(log(dateSpan^2+1)) (139,636-140,635)
- uID → log(time diff to obj start+1) (140,636-140,636)
- eID → dateVec diff stats (140,637-140,649)

2.6 F5

- Course ID - One-hot-encoded course_id
- Source time counts by enrollment - The log count of each source type per day for each enrollment
- Source time counts by course id - The log count of each source type per day for each course id
- Event time counts by enrollment - The log count of each event type per day for each enrollment
- Event time counts by course id - The log count of each event type per day for each course id

2.7 F6

Features generated by Jeong-Yoon Lee are as follows:

- User ID (20,113) - One-hot-encoded username. Usernames appearing less than 100 times in training log data are grouped together as one user ID.
- Course ID (39) - One-hot-encoded course_id.
- Source Event (10) - One-hot-encoded combination of source and event.
- Object ID (3,554) - One-hot-encoded object. Objects appearing less than 100 times in training log data are grouped together as one object ID.
- Count (1) - Number of log entries for an hour_id.
- Object Category (6) - Number of log entries with an object category for an enrollment_id.
- Number of Children Objects (7) - One-hot-encoded total number of object's children for an enrollment_id.
- Object Timespan (10) - One-hot-encoded timespan in days between object's start date and last day of the class
- Day of Class (30) - One-hot-encoded day of the class
- Week of Class (4) - One-hot-encoded week of the class
- End Month of Class (7) - One-hot-encoded end month of the class
- Object Started in Dropout Period (2) - Binary variable that is 1 if object started after but before 10 days after last day of the class and 0 otherwise.

2.8 F7

F1 + magic feature

3. LEARNING FRAMEWORK

Our final AUC score of 0.90918 results from a complex pipeline from raw data to final score. Every part of that pipe needs to be (sub-)optimal implemented by our team to get the best score at the end. The first part "feature design" is the most important one and needs expertise, experience and of course a bit luck to capture all signals in the data.

3.1 Single Models

something like this.

- Logistic Regression (LR)
- Kernel Ridge Regression (KRR)
- Factorization Machine (FM)

- Neural Networks (NN)
- Random Forests (RF)
- Extra Trees (ET)
- Gradient Boosting Machine (GBM)

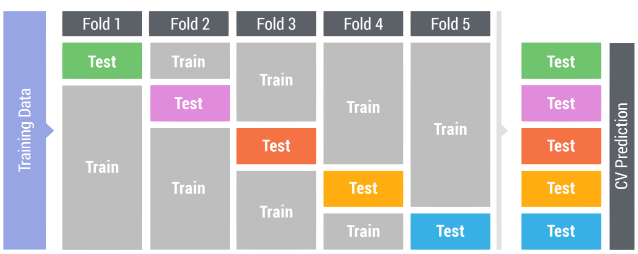
3.2 Single Models

- Model 1: RandomForest(R). Dataset: X
- Model 2: Logistic Regression(scikit). Dataset: Log(X+1)
- Model 3: Extra Trees Classifier(scikit). Dataset: Log(X+1) (but could be raw)
- Model 4: KNeighborsClassifier(scikit). Dataset: Scale(Log(X+1))
- Model 5: libfm. Dataset: Sparse(X). Each feature value is a unique level.

F2

3.3 Model Validation

Figure 1: 5-fold CV



We use stratified 5-fold cross validation (CV) for model validation and ensemble. Training data are split into five folds while the sample size and dropout rate are preserved across folds.

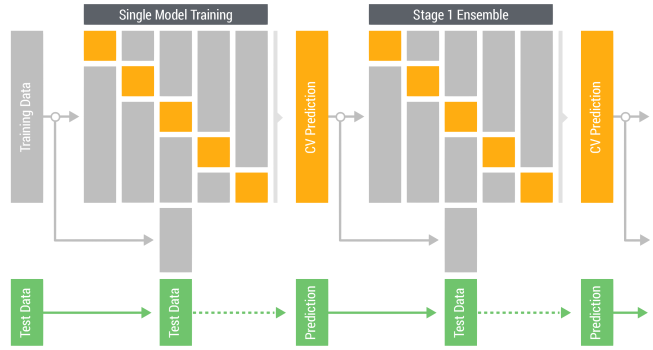
For validation, each of single and ensemble models is trained five times. Each time, one fold is held out and the remaining four folds are used for training. Then, predictions for the hold-out folds are combined and form the model's CV prediction. CV predictions are used as inputs for ensemble model training as well as validation score calculation.

For test, each of single and ensemble models is retrained with whole training data. Then predictions for test data are used as inputs for ensemble prediction as well as for submission.

3.4 Multi-Stage Ensemble

We use the multi-stage ensemble with stacked generalization [4, 3] to blend predictions of multiple models. As shown in Figure 2, in each stage, we train ensemble models with 5-fold CV, and use the CV and test predictions of models in the previous stage as inputs. Then, we pass the CV and test predictions of the ensemble models to the next stage as inputs.

Figure 2: 5-fold CV stacked generalization ensemble



4. FINAL SOLUTION

Our final AUC score of 0.90918 results from a complex pipeline from raw data to final score. Every part of that pipe needs to be (sub-)optimal implemented by our team to get the best score at the end. The first part "feature design" is the most important one and needs expertise, experience and of course a bit luck to capture all signals in the data.

We train 64 single models with 7 algorithms and 7 feature sets.

We trained 15 stage-I ensemble classifiers with different subsets of CV predictions of 64 single classifiers.

At KDD Cup 2015, GBM outperforms other algorithms. Our top 8 single models as well as top 2 stage-1 ensemble models are trained with GBM.

We trained 2 stage-II ensemble classifiers with different subsets of CV predictions of 15 stage-I ensemble classifiers.

We trained a stage-III ensemble classifier with CV predictions of 5 classifiers: 1 stage-II ensemble, 3 stage-I ensemble, and 1 individual classifiers.

Performance improvement diminishes as we add more ensemble stages. The stage-1 ensemble improves the CV AUC score by XXXX from 0.906721 to 0.907688. The stage-2 ensemble improves the CV AUC score by XXXX to 0.907968. The stage-3 ensemble improves the CV AUC score by XXXX to 0.908194.

We choose subsets of predictions from the previous stage for ensemble model training,

A linear combination of the 5 models from table 4 results in train AUC=0.908072 and accuracy=0.887334. Which leads to 0.90910 public leaderboard score. By adding 39 courseID correction factors train AUC=0.908194 and public score improved to 0.90918.

ID	Model	Type	5-CV	Weight
S1	xgb_rf.ko_new_feat	Single	0.906721	1.1703
E4	at.esb50v2+ko	Stage I	0.907878	1.9626
E8	esb58v5+magic.dae+nn	Stage I	0.907567	0.7871
E18	et_esb58v5_rank	Stage I	0.906207	0.4580
E2	lr_forward_0.01_esb.esb15v3	Stage II	0.907968	1.6146

5. CONCLUSIONS

Our final AUC score of 0.90918 results from a complex pipeline from raw data to final score. Every part of that pipe needs to be (sub-)optimal implemented by our team to

Figure 3: 5-fold CV vs. public leaderboard AUC scores

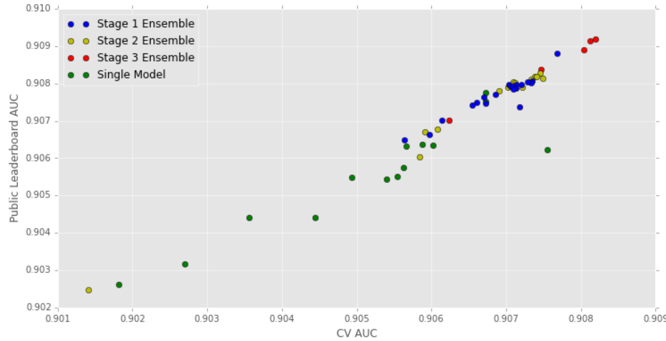
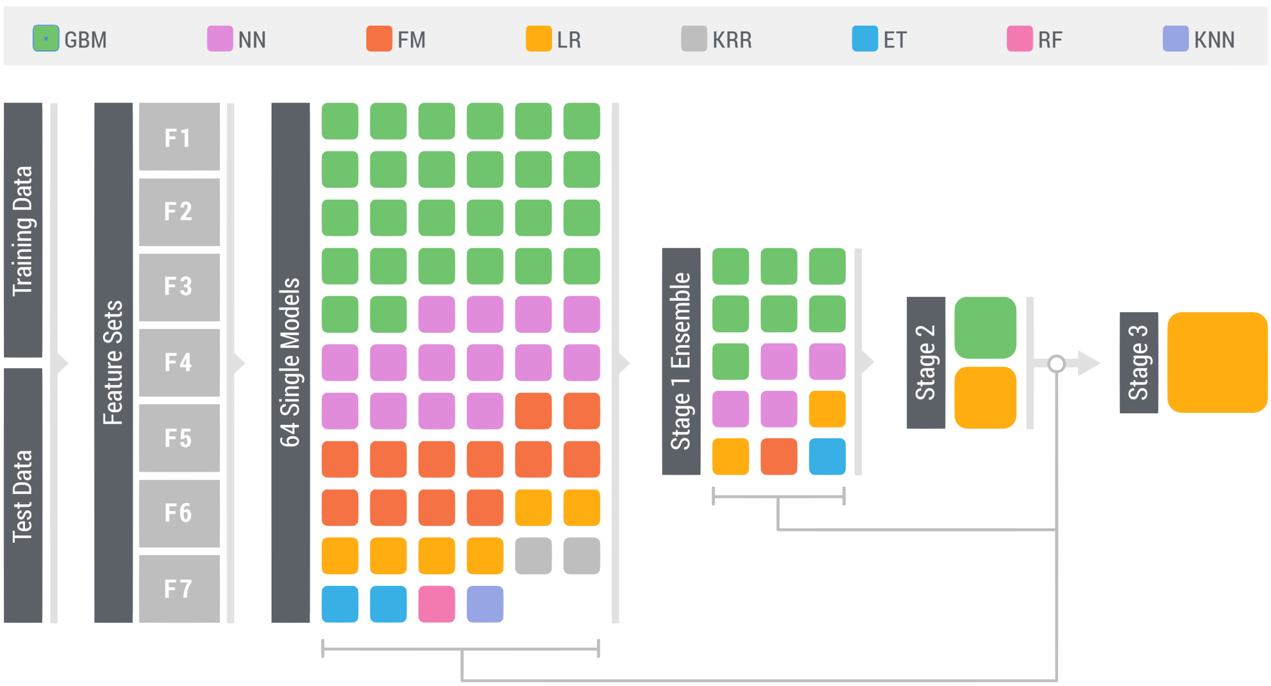


Figure 4: End-to-end pipeline for the final solution



get the best score at the end. The first part "feature design" is the most important one and needs expertise, experience and of course a bit luck to capture all signals in the data.

6. ADDITIONAL AUTHORS

Additional authors: Xiaocong Zhou (Tsinghua University, email: `infinitezxc@gmail.com`), Song Chen (AIG, email: `song.chen@aig.com`) and Peng Yan (NetEase Youdao, email: `yanpeng@rd.netease.com`).

7. REFERENCES

[1] http://kddcup2015.com/submission-data.html.

[2] L. Perna, A. Ruby, R. Boruch, N. Wang, J. Scull, C. Evans, and S. Ahmad. The life cycle of a million mooc users. In *Presentation at the MOOC Research Initiative Conference*, 2013.

[3] K. M. Ting and I. H. Witten. Issues in stacked generalization. *J. Artif. Intell. Res.(JAIR)*, 10:271–289, 1999.

[4] D. H. Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

No.	Description
1	Course_id encoded by 1-of-N coding
2	Number of requests by an enrollment_id
3	Number of unique object by an enrollment_id
4	Number of unique problem object of event by an enrollment_id
5	Number of active days by an enrollment_id
6	Number of active hours by an enrollment_id
7	Time of first access in hours by an enrollment_id
8	Time of last access in hours by an enrollment_id
9	Number of enrollments by an username
10	Number of requests by an username
11	Number of unique objects by an username
12	Number of unique problem object of event by an username
13	Number of active days by an username
14	Number of active hours by an username
15	Time of first access in hours by an username
16	Time of last access in hours by an username
17	Time of first problem access in hours by an username
18	Time of last problem access in hours by an username
19	For each course, number of requests by an username
20	For each course, number of unique object by an username
21	For each course, number of unique problem object by an username
22	For each course, number of active days by an username
23	For each course, number of active hours by an username
24	For each course, time of first access in hours
25	For each course, time of last access in hours
26	Number of enrollment_ids during 10 days after the end date of course by an username
27	For each course, number of access logs during 10 days after the end date of course by an username
28	For each course, number of unique objects during 10 days after the end date of course by an username
29	For each course, number of unique problem objects during 10 days after the end date of course by an username
30	For each course, number of active hours during 10 days after the end date of course by an username
31	For each course, difference between first and last access during 10 days after the end date of course by an username
32	For each course, time of first access in hours during 10 days after the end date of course by an username
33	For each course, time of last access in hours during 10 days after the end date of course by an username
34	For each course, time of first access to an problem object in hours during 10 days after the end date of course by an username
35	For each course, time of last access to an problem object in hours during 10 days after the end date of course by an username
36	Number of enrollment_ids during 1 day after the end date of course by an username
37	For each course, number of access logs during 1 day after the end date of course by an username
38	For each course, number of unique objects during 1 day after the end date of course by an username
39	For each course, number of unique problem objects during 1 day after the end date of course by an username
40	For each course, number of active hours during 1 day after the end date of course by an username
41	For each course, difference between first and last access during 1 day after the end date of course by an username
42	For each course, time of first access in hours during 1 day after the end date of course by an username
43	For each course, time of last access in hours during 1 day after the end date of course by an username
44	For each course, time of first access to an problem object in hours during 1 day after the end date of course by an username
45	For each course, time of last access to an problem object in hours during 1 day after the end date of course by an username
46	For each days of the course, which date is provided in date.csv, number of unique active courses by an username
47	For each 10 days after the end date of the course, number of active enrollment_id, which target variables are 1 in the training set, enrolled by
48	For each 10 days after the end date of the course, number of active enrollment_id, which target variables are 0 in the training set, enrolled by
49	For each 10 days after the end date of the course, number of active enrollment_id (in this case, days between last access and the end date of t
50	For each 10 days after the end date of the course, number of active enrollment_id (in this case, days between last access and the end date of t
51	For each 14 days before the end date of the coruses, number of active enrollment_id, which target variables are 1 in the training set, enrolled
52	For each 14 days before the end date of the coruses, number of active enrollment_id, which target variables are 0 in the training set, enrolled
53	For each 14 days before the end date of the coruses, number of active enrollment_id (in this case, days between last access and the end date o
54	For each 14 days before the end date of the coruses, number of active enrollment_id (in this case, days between last access and the end date o

Table 1: List of features generated by Song and Kohei.

No.	Description
1	act counts
2	hourset length in last 2 days
3	last month
4	max absent days
5	day set length
6	hour set length
7	average hours per day
8	event wiki counts
9	event discussion counts
10	event access counts
11	event video counts
12	event problem counts
13	obj chapter not visited
14	obj chapter visited ratio
15	obj video not visited
16	obj video visited ratio
17	obj problem not visited
18	obj problem visited ratio
19	obj set length
20	total time span
21	days from last act to course end
22	course drop rate
23	number of courses enrolled
24	min days between first visit and next course begin
25	min days between 10 days after last visit and next course begin
26	min days between last visit and next course end
27	min days between previous course end and last visit
28	min days between 10 days after current course end and next course begin
29	min days between 10 days after current course end and next course end
30	min days between current course end and next visit
31	number of active days between last visit and course end
32	number of active days in 10 days after course end
33	number of courses ended after current course end

Table 2: List of features generated by Peng and Xiaocong.

ID	Model	Feature	5-CV	Public Leaderboard
S1	xgb_rf_ko_new_feat	F2 + F3 + F6 + F7	0.906721	0.907765
S2	ko_v83	F2 + F1 + F6 + F7	0.906729	0.907525
S3	bag_10_xgb_rf.xiv	F2 + F1 + F3	0.905875	0.906361
S4	xgb_rf.xvi	F2 + F3 + F6	0.905543	0.905516
S5	xgb_rf.xix	F2 + F1 + F3 + F5	0.905356	-
S6	rw_sk_xgb	F2 + F1	0.905312	-
S7	xgb_rf.xv	F2 + F3	0.904935	0.905480
S8	xgb_rw_sk	F2 + F1	0.904914	-
S9	mjahrer.feat.tam+rw+sk+tam2+sk2+azure.dae+nn	F2 + F1 + F3 + F4	0.904235	-
S10	mjahrer.featmjahrer+Tam+RW+KoheiSong.nn	F2 + F1 + F3 + F4	0.903736	-
S11	mjahrer.featTam+RW+KoheiSong.dae+nn	F2 + F1 + F3	0.903669	-
S12	pred_train_sk_feature_ffm_rw	F1 + F2	0.903560	0.904411
S13	mjahrer.featTam+RW+KoheiSong.nn	F2 + F1 + F3	0.903428	-
S14	xg_400_4_0.05_feature_rw	F2 + F6	0.903385	-
S15	kohei_song	F1	0.902918	-
S16	xgb_cv_rw	F2	0.902287	-
S17	ffm_cv_rw	F2	0.901983	-
S18	mjahrer.dae+nn.RWfeat	F2	0.901846	0.902614
S19	mjahrer.featmjahrer+Tam+RW+KoheiSong.dae+krr	F2 + F1 + F3 + F4	0.901522	-
S20	mjahrer.feat.tam+rw+sk+tam2+sk2+azure.BIN	F2 + F1 + F3	0.900982	-
S21	xg_400_4_0.05_feature_sk	F1	0.900906	-
S22	xgb_rf.xiv	F3	0.899239	-
S23	xgb_rf.xiv	F3	0.899167	-
S24	xgb_rf.xiv	F3	0.898969	-
S25	xgb_rf.xiv	F3	0.898890	-
S26	xgb.xiv	F3	0.898749	-
S27	libfm_100_4_0.002_feature_rw_v2	F2 + F6	0.898308	-
S28	xgl_500_0.5_10_10_feature_rw_v2	F2 + F6	0.897968	-
S29	xgb_val.xiii	F3	0.897912	-
S30	nn_20_16_0.01_feature_rw_v2	F2 + F6	0.897143	-
S31	mb_nn_50_20_feat19	F2 + F5	0.896748	-
S32	sm_rw_sk_tam	F2 + F1 + F3	0.896435	-
S33	ffm_30_20_0.01_feature_rw_v2	F2 + F6	0.896160	-
S34	xg_400_4_0.05_feature_tam	F3	0.895754	-
S35	ConfigAMLKRR	F4	0.894524	-
S36	gbm.xiii	F3	0.893507	-
S37	xg_600_4_0.05_feature10	F6	0.892364	-
S38	xg_600_4_0.05_feature9	F6	0.892253	-
S39	ConfigAML	F4	0.891217	-
S40	lr_0.01_feature_tam	F3	0.890580	-
S41	ConfigAMLCUDAPreModel	F4	0.890565	-
S42	ffm_30_20_0.01_feature_tam	F3	0.888418	-
S43	libfm_100_4_0.002_feature_tam	F3	0.888381	-
S44	rf.xiii	F3	0.887583	-
S45	et_1000_20_feature_tam	F3	0.887768	-
S46	ffm_30_20_0.01_feature9	F6	0.887116	-
S47	libfm_100_4_0.002_feature9	F6	0.886866	-
S48	ConfigAML	F4	0.886705	-
S49	nn_20_16_0.01_feature9	F6	0.886109	-
S50	xg_400_4_0.05_feature6	F6	0.885184	-
S51	xg_400_4_0.05_feature3	F6	0.885124	-
S52	ffm_20_20_0.01_feature6	F6	0.885037	-
S53	ffm_20_20_0.01_feature3	F6	0.884697	-
S54	xg_400_4_0.05_feature_mj	F4	0.882441	-
S55	et_1000_20_feature_rw_v2	F2 + F6	0.881539	-
S56	libfm_100_8_0.01_feature6	F6	0.880878	-
S57	libfm_100_8_0.01_feature3	F6	0.880366	-
S58	nn_20_16_0.005_feature3	F6	0.880219	-
S59	ConfigAMLKNN	F4	0.877652	-
S60	nn_20_16_0.005_feature5	F6	0.876905	-
S61	lr_0.01_feature_mj	F4	0.821332	-
S62	lr_0.01_feature9	F6	0.804150	-
S63	lr_0.01_feature3	F6	0.802138	-
S64	lr_0.01_feature6	F6	0.800225	-

ID	Stage	Model	5-CV	Public Leaderboard
E1	III	subBlend_0712v2	0.908194	0.909181
E2	II	lr_forward_0.01_esb.esb15v3	0.907968	-
E3	II	xgl_10_0.01_10_10_esb.esb11v6	0.907379	0.908187
E4	I	at.esb50v2+ko	0.907878	-
E5	I	xg_sk_1800_5_0.004_esb51_rank	0.907734	
E6	I	lr_forward_0.01_esb51_rank_norm	0.907716	-
E7	I	song_train_xgb_esb58v5_ko	0.907668	0.908796
E8	I	esb58v5+magic.dae+nn	0.907567	-
E9	I	esb58v3.trn.final	0.907353	0.908060
E10	I	trn.esb56.blend.at	0.907283	0.908043
E11	I	ConfigAMLCUDAPreModel	0.907076	-
E12	I	xg_sk_1800_5_0.004_esb56_rank_4	0.907036	0.907977
E13	I	esb55.dae+nn	0.906956	-
E14	I	lr_0.01_esb51_rank_norm	0.906746	-
E15	I	nn	0.906714	-
E16	I	xgb_rf_esb55.xix	0.906689	-
E17	I	libfm	0.906537	-
E18	I	et_esb58v5_rank	0.906200	-