Feature Engineering and Classifier Ensemble for KDD Cup 2010

Chih-Jen Lin
Department of Computer Science
National Taiwan University

Joint work with HF Yu, HY Lo, HP Hsieh, JK Lou, T McKenzie, JW Chou, PH Chung, CH Ho, CF Chang, YH Wei, JY Weng, ES Yan, CW Chang, TT Kuo, YC Lo, PT Chang, C Po, CY Wang, YH Huang, CW Hung, YX Ruan, YS Lin, SD Lin and HT Lin

Outline

- Introduction
- Course at NTU
- Initial Approaches and Some Settings
- Sparse Features and Linear Classification
- Condensed Features and Random Forest
- Ensemble and Final Results
- Discussion and Conclusions

Outline

KDD Cup

- Annual data mining and knowledge discovery competition
- Organized by ACM special interest group on knowledge discovery and data mining
- 1997-present
- Now considered the most prestigious data mining competition

KDD Cup 2010

- Educational data mining competition
  `https://pslcdatashop.web.cmu.edu/KDDCup/`
- Predicting student algebraic problem performance given information regarding past performance
- Training data: summaries of the logs of student interaction with intelligent tutoring systems
- Two data sets: algebra_2008_2009 and bridge_to_algebra_2008_2009.
- We refer to them as A89 and B89, respectively.

KDD Cup 2010 (Cont'd)

- Each data set: logs for a large number of interaction steps
- A89: 8,918,055 steps; B89: 20,012,499 steps

Log Fields

- student ID
- problem hierarchy including step name, problem name, unit name, section name
- knowledge components (KC) used in the problem
- number of times a problem has been viewed

Some log fields are only available in the training set:

- whether the student was correct on the first attempt for this step (CFA)
- number of hints requested (hint)
- step duration information.

Log Fields (Cont'd)

Hierarchy: step ⊂ problem ⊂ section ⊂ unit

Unit

CTA1_02 CTA1_01 ES_01 UNIT-CONVERSIONS-ONE-STEP

Section

CTA1_02-4 CTA1_01-4 ES_01-11
UNIT-CONVERSIONS-ONE-STEP-2

Problem

EG27 -5=-y PROP03 RATIO4-135 L2FB14B

Step

Series1AddPoint1 5=-y*(-1) ValidEquations R5C2

Log Fields (Cont'd)

KC examples:

```
KC subskills:
 Using simple numbers~~Find Y, any form~~Find Y
 Enter unit conversion
 Entering a given~~Enter given, reading words
 Entering a given~~Enter given, reading numeral
KC KTracedSkills:
 Identifying units-1
 Convert linear units-1~~Convert decimal units
 Select form of one with denominator of one-1
 Enter unit conversion-1
```

Generation of Training/Testing Data

- Testing data: generated by randomly drawing a problem from a unit
- Problems before are used as training and after are discarded.

A unit of problems

problem 1 $\in T$

problem 2 $\in T$

$\vdots$

problem $i \in \tilde{T}$

problem i+1: not used

$\vdots$

T: training $\tilde{T}$: testing

Competition Goal

- Predict CFA
- 0 (i.e., incorrect on the first attempt) or 1
- Training: CFA is available to participants
- A testing set of unknown CFA is left for evaluation
- Evaluation criterion: root mean squared error (RMSE)

$$\sqrt{\frac{\|\mathbf{p}-\mathbf{y}\|^2}{l}}$$

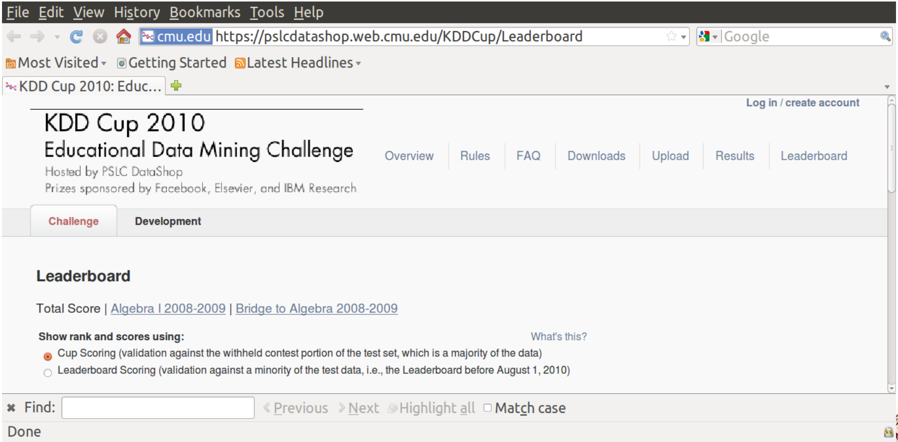
l: # testing data, $\mathbf{p} \in [0,1]^l$: predictions, $\mathbf{y} \in \{0,1\}^l$: true answers

KDD Cup 2010 Schedule

- April 1: Registration opens at 2pm EDT, development data sets available
- April 19: Competition starts at 2pm EDT, challenge data sets available
- June 8: Competition ends at 11:59pm EDT
- June 14: Fact sheet and team composition info due by 11:59pm EDT
- June 21: Winners announced
- July 25: Workshop at ACM KDD 2010

Leaderboard

Based on results of a “unidentified” portion of testing data



Leaderboard (Cont'd)

File Edit View History Bookmarks Tools Help

cmu.edu https://pslcdatashop.web.cmu.edu/KDDCup/Leaderboard

Most Visited Getting Started Latest Headlines

KDD Cup 2010: Educ...

Leaderboard Scoring (validation against a minority of the test data, i.e., the Leaderboard before August 1, 2010)

Number of rows: 3925

Rows per page 10

1-10 of 3925 First | Back | **Next** | **Last**

Overall Rank	Individual/Team Name	Algebra I 2008-2009	Bridge to Algebra 2008-2009	Total Score	Date
1	NTU	0.274311	0.271157	0.272734	2010-06-08 23:46:36
2	NTU	0.274309	0.271162	0.272736	2010-06-08 13:28:24
3	NTU	0.274311	0.271163	0.272737	2010-06-08 22:30:56
4	NTU	0.274311	0.271163	0.272737	2010-06-08 22:32:22
5	NTU	0.274311	0.271163	0.272737	2010-06-08 13:09:07
6	NTU	0.274311	0.271163	0.272737	2010-06-08 22:25:29
7	NTU	0.274311	0.271163	0.272737	2010-06-08 13:43:21
8	NTU	0.274311	0.271163	0.272737	2010-06-08 22:18:28
9	NTU	0.274311	0.271163	0.272737	2010-06-08 22:22:56

Find: Previous Next Highlight all Match case

Done

Outline

- Introduction
- Course at NTU
- Initial Approaches and Some Settings
- Sparse Features and Linear Classification
- Condensed Features and Random Forest
- Ensemble and Final Results
- Discussion and Conclusions

Course at NTU

- At National Taiwan University, we organized a course for KDD Cup 2010
- Course page: `http://www.csie.ntu.edu.tw/~cjlin/courses/dmcase2010/`
- Wiki: used to record progress

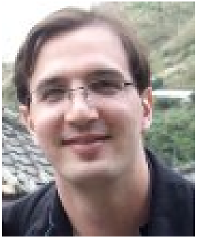
Team Members

- Three instructors, two TAs, 19 students and one RA
- 19 students split to six sub-teams

  Named by animals

  Armyants, starfish, weka, trilobite, duck, sunfish
- Every week each team reports progress

Armyants




麥陶德 (Todd G. McKenzie), 羅經凱 (Jing-Kai Lou) and 解巽評 (Hsun-Ping Hsieh)

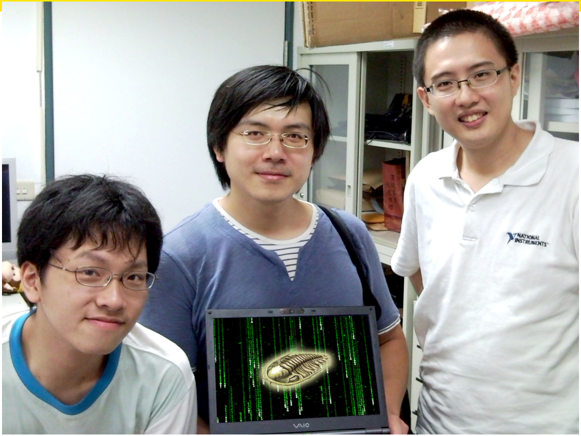
Starfish

Chia-Hua Ho (何家華), Po-Han Chung (鐘博翰), and Jung-Wei Chou (周融瑋)

Weka

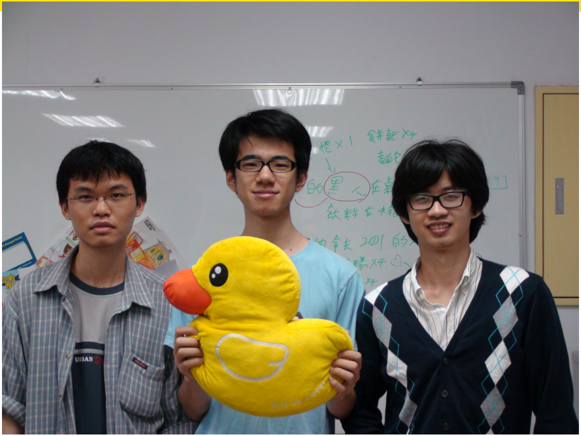
Yin-Hsuan Wei (魏吟軒), En-Hsu Yen (嚴恩勗), Chun-Fu Chang (張淳富) and Jui-Yu Weng (翁睿妤)

Trilobite



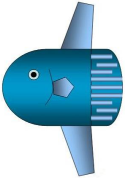
Yi-Chen Lo (羅亦辰), Che-Wei Chang (張哲維) and Tsung-Ting Kuo (郭宗廷)

Duck



Chien-Yuan Wang (王建元), Chieh Po (柏傑), and Po-Tzu Chang (張博詞).

Sunfish

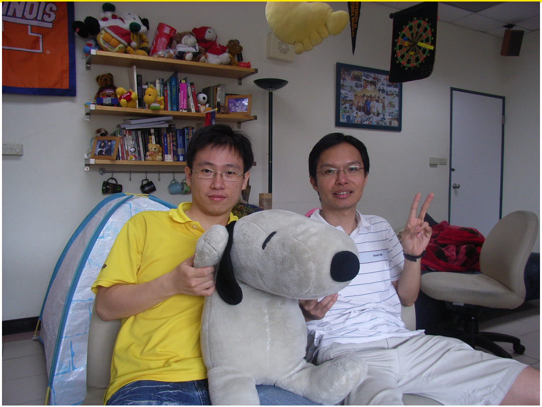


Yu-Xun Ruan (阮昱勳), Chen-Wei Hung (洪琛洧) and Yi-Hung Huang (黃曳弘)

Tiger (RA)

Yu-Shi Lin (林育仕)

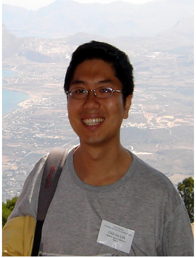
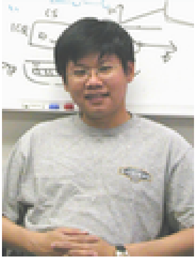
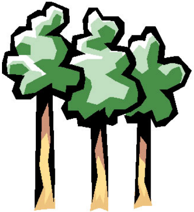
Snoopy (TAs)



Hsiang-Fu Yu (余相甫) and Hung-Yi Lo (駱宏毅)

Snoopy and Pikachu are IDs of our team in the final stage of the competition

Instructors

林智仁 (Chih-Jen Lin), 林軒田 (Hsuan-Tien Lin) and 林守德 (Shou-De Lin)

Outline

Initial Thoughts and Our Approach

We suspected that this competition would be very different from past KDD Cups

- Domain knowledge seems to be extremely important for educational systems
- Temporal information may be crucial

At first, we explored a temporal approach

- We tried Bayesian networks
- But quickly found that using a traditional classification approach is easier

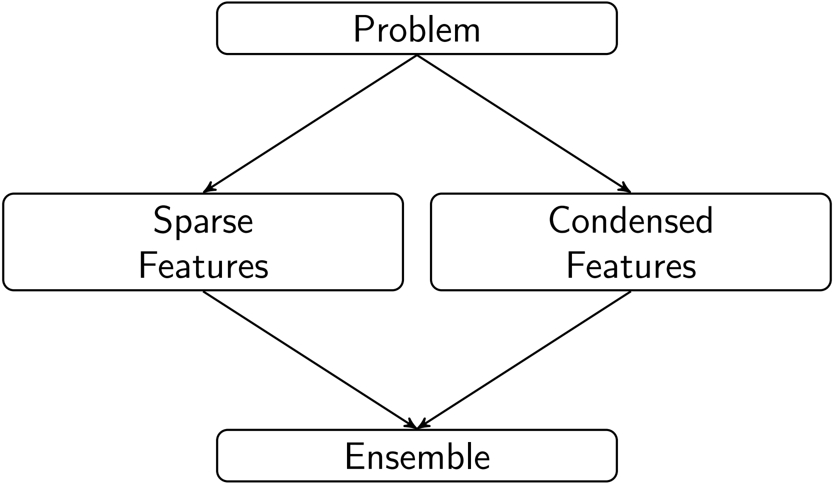
Initial Thoughts and Our Approach (Cont'd)

Traditional classification:

- Data points: independent Euclidean vectors
- Suitable features to reflect domain knowledge and temporal information

Domain knowledge, temporal information: important, but not as extremely important as we thought in the beginning

Our Framework



Validation Sets

- Avoid overfitting the leader board
- Standard validation
  $\Rightarrow$ ignore time series
- Our validation set: last problem of each unit in training set
- Simulate the procedure to construct testing sets

A unit of problems

- problem $1 \in V$
- problem $2 \in V$
- $\vdots$
- last problem $\in \tilde{V}$

V: internal training
$\tilde{V}$: internal validation

- In the early stage, we focused on validation sets

Validation Sets (Cont'd)

A89: algebra_2008_2009
B89: bridge_to_algebra_2008_2009

	A89	B89
Internal training	8,407,752	19,264,097
Internal validation	510,303	748,402
External training	8,918,055	20,012,499
External testing	508,913	756,387

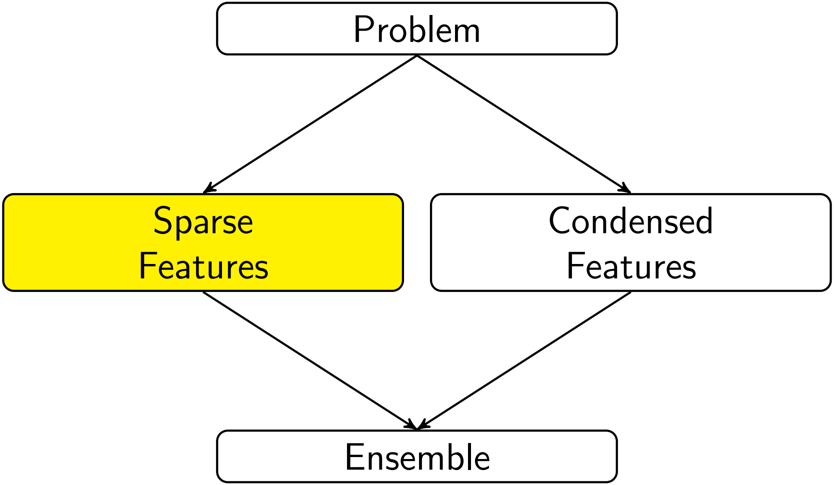
- In the early stages, we focused on validation sets
- Each sub-team submits to the leader board only once per week

Validation Sets (Cont'd)

- This avoid overfitting the leaderboard
- Of course in the end, many teams slightly violated the rule to submit more results in a week

Outline

- Introduction
- Course at NTU
- Initial Approaches and Some Settings
- Sparse Features and Linear Classification
- Condensed Features and Random Forest
- Ensemble and Final Results
- Discussion and Conclusions


Problem
Sparse Features
Condensed Features
Ensemble

Basic Sparse Features

Categorical: expanded to binary features

- student, unit, section, problem, step, KC
- For example, 3,310 students in A89 $\Rightarrow$ feature vector then contains 3,310 binary features to indicate the student who finished the step.

Numerical: scaled by $\log(1 + x)$

- opportunity value, problem view
- original range of opportunity in $[1, 1504]$, problem view in $[1, 18]$ for A89
- original range of opportunity in $[1, 2402]$, problem view in $[1, 29]$ for B89
- We have tried other scaling methods (e.g., linear scaling)

Basic Sparse Features (Cont'd)

A89: algebra_2008_2009

B89: bridge_to_algebra_2008_2009

Data	stud.	unit	sec.	prob.	step	KC
A89	3,310	42	165	$192,811\times 2$	725,652	$2,097\times 2$
B89	6,043	50	186	$53,375\times 2$	129,349	$1,699\times 2$

- Number of features: 1M for A89, 200K for B89
- prob.: problem and problem view
- KC: KC and opportunity

Basic Sparse Features (Cont'd)

Results:

RMSE (leader board)	A89	B89
Basic sparse features	0.2895	0.2985
Best leader board	0.2759	0.2777

- Five of six student sub-teams use variants of this approach
- From this basic set, we add more features

Extensions from Basic Sparse Features

- Different scaling methods
- Slightly different ways to generate features
- Slightly different subsets of features
- Different regularization (L1 and L2) for classification

We will discuss some in detail

Feature Combination

- Due to large training size, nonlinear classifiers (e.g., kernel SVM) are not practical
- Linear classifier viable, but not exploiting possible feature dependence
- Following polynomial mapping in kernel methods or bigram/trigram in NLP, we use feature combinations to indicate relationships.
- We manually identify some useful combinations for experiments

Feature Combination (Cont'd)

- Example: hierarchical information
  (student name, unit name), (unit name, section name), (section name, problem name) and (problem name, step name)
- We have also explored combinations of higher-order features (i.e., more than two)
- We released two data sets using feature combinations at
  `http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`
- We thank Carnegie Learning and Datashop for allowing us to release them

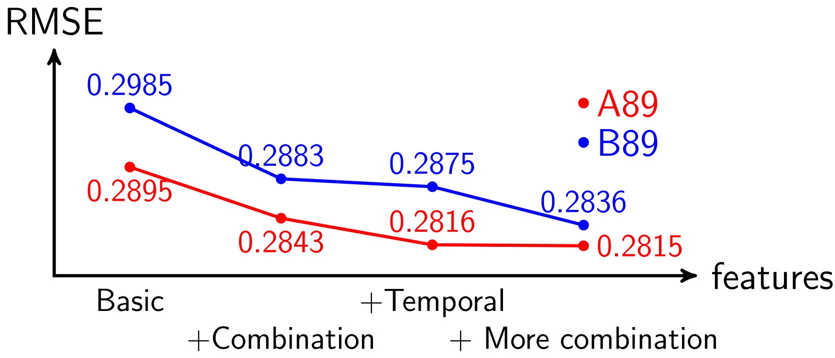
Temporal Information

- Learning is a process of skill-improving over time
- Temporal information should be taken into consideration.
- We considered a simple and common approach:

  For each step, step name and KC values from the previous few steps were added as features.

Feature Combination and Temporal Information

Leaderboard results



Feature Combination and Temporal Information (Cont'd)

- Feature combinations very useful for B89
- Temporal features more useful for A89
- More features improve RMSE; but improvement less dramatic

  Information already realized by earlier feature combinations

Details of Features

+Combination	(student name, unit name), (unit name, section name), (section name, problem name), (problem name, step name), (student name, unit name, section name), (unit name, section name, problem name), (section name, problem name, step name), (student name, unit name, section name, problem name) and (unit name, section name, problem name, step name)
+Temporal	Given a student and a problem, add KCs and step name in each previous three steps as temporal features.
+More combination	(student name, section name), (student name, problem name), (student name, step name), (student name, KC) and (student name, unit name, section name, problem name, step name)

Number of Features

Features	A89	B89
Basic	1,118,985	245,776
+Combination	6,569,589	4,083,376
+Temporal	8,752,836	4,476,520
+More combination	21,684,170	30,971,151

Important Feature Combinations

#features	A89	B89
Basic	0.2895	0.2985
+ (problem name, step name)	0.2851	0.2941
+ (student name, unit name)	0.2881	0.2942
+ (problem name, step name) and (student name, unit name)	0.2842	0.2898
+ Combination	0.2843	0.2883

- (problem name, step name) and (student name, unit name) are very useful

Other Feature Generations

- We tried many other ways
- We will discuss some of them
- They may be less effective than feature combinations mentioned earlier

Knowledge Component Feature

Originally using binary features to indicate if a KC appears. An alternative way:

Each token in KC as a feature

- "Write expression, positive one slope" similar to "Write expression, positive slope"
- Use "write," "expression," "positive" "slope," and "one" as binary features
- Performs well on A89 only

Grouping Similar Names

- Two step names "$-18 + x = 15$" and "$5 + x = -39$" differ only in their numbers.
- For problem name and step name, we tried to group similar names together
- By replacing numbers with a symbol, they become the same string and hence the same step name
- Number of features reduced without deteriorating the performance

Training via Linear Classification

- Large numbers of instances and features
- The largest number of features used is 30,971,151

	#instances	#features
A89	8,918,055	$\geq$ 20M
B89	20,012,499	$\geq$ 30M

- Impractical to use nonlinear classifiers
- Use LIBLINEAR developed at National Taiwan University (Fan et al., 2008)
- We consider logistic regression instead of SVM
- Training time: about 1 hour for 20M instances and 30M features (B89)

Training via Linear Classification (Cont'd)

- Logistic regression: CFA as label y_i

$$y_i = \begin{cases} 1 & \text{if CFA} = 1, \\ -1 & \text{if CFA} = 0, \end{cases}$$

- Assume training set includes $(\mathbf{x}_i, y_i), \; i = 1, \ldots, l$.
- Logistic regression assumes the following probability model:

$$\mathcal{P}(y \mid \mathbf{x}) = \frac{1}{1 + \exp(-y\mathbf{w}^T\mathbf{x})}.$$

Training via Linear Classification (Cont'd)

- Regularized logistic regression solves

$$\min_{\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l} \log\left(1 + e^{-y_i\mathbf{w}^T\mathbf{x}_i}\right) \qquad (1)$$

  $\mathbf{w}$: weight vector of the decision function, $\mathbf{w}^T\mathbf{w}/2$: L2-regularization term, and C: penalty parameter.
- C: often decided by validation. We used $C = 1$ most of the time.

Training via Linear Classification (Cont'd)

- L2 regularization: a dense vector **w**; we have also considered L1 regularization to obtain a sparse **w**:

$$\min_{\mathbf{w}} \quad \|\mathbf{w}\|_1 + C \sum_{i=1}^{l} \log\left(1 + e^{-y_i \mathbf{w}^T \mathbf{x}_i}\right). \qquad (2)$$

- Once **w** is obtained, we submitted either

$$y = \begin{cases} 1 & \text{if } \mathbf{w}^T \mathbf{x} \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

or probability values

$$1/(1 + \exp(-\mathbf{w}^T \mathbf{x}))$$

Training via Linear Classification (Cont'd)

Using probability values gives a smaller RMSE than using 1/0

- Assume the true label is 0.
- Wrong prediction: errors using label/probability are 1 and $(1 - p_1)^2$

  $p_1 \geq 0.5$: predicted probability
- Correct prediction: errors are 0 and p^2, respectively.

  $p_2 \leq 0.5$: predicted probability

Training via Linear Classification (Cont'd)

- Quadratic function is increasing in $[0, 1]$,
- Gain of reducing 1 to $(1-p)^2$ is often larger than loss of increasing 0 to p^2.
- Example:

	p	error	label	error
Wrong	0.75	0.5625	1	1
Correct	0.25	0.0625	0	0

Training via Linear Classification (Cont'd)

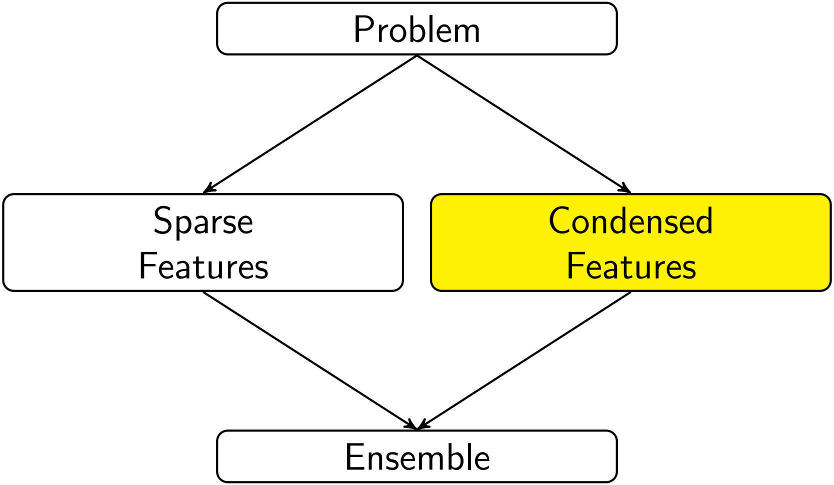
- We also checked linear support vector machine (SVM) solvers in LIBLINEAR
- Result was slightly worse than logistic regression.

Result Using Sparse Features

Leader board results:

	A89	B89
Basic sparse features	0.2895	0.2985
Best sparse features	0.2784	0.2830
Best leader board	0.2759	0.2777

Outline


Problem
Sparse Features
Condensed Features
Ensemble

Condensed Features

Categorical feature $\Rightarrow$ numerical feature

- Use correct first attempt rate (CFAR). Example: a student named sid:

$$\text{CFAR} = \frac{\#\ \text{steps with student} = \text{sid and CFA} = 1}{\#\ \text{steps with student} = \text{sid}}$$

- CFARs for student, step, KC, problem, (student, unit), (problem, step), (student, KC) and (student, problem)

Condensed Features (Cont'd)

Temporal features: the previous ≤ 6 steps with the same student and KC

- An indicator for the existence of such steps
- Average of CFAs
- Average hints (up to six depending on the availability)

Other temporal features:

- When was a step with the same student name and KC be seen?
- Binary features to model four levels:

  Same day, 1-6 days, 7-30 days, $>$ 30 days

Condensed Features (Cont'd)

Opportunity and problem view:

- First scaled by

$$\frac{x}{x+1}$$

- Then linearly scaled to $[0, 1]$

Total 17 condensed features

- Eight CFARs
- Seven temporal features
- Two scaled numerical features for opportunity and problem view.

Training by Random Forest

- Due to a small number of features, we could try several classifiers via Weka (Hall et al., 2009)
- To save training time, we considered a subset of training data and split the classification task into several independent sets according to unit name.
- That is, for each unit name, we collected the last problem of each unit to form its training set.
- In testing, we checked the testing point's unit name to know which model to use.

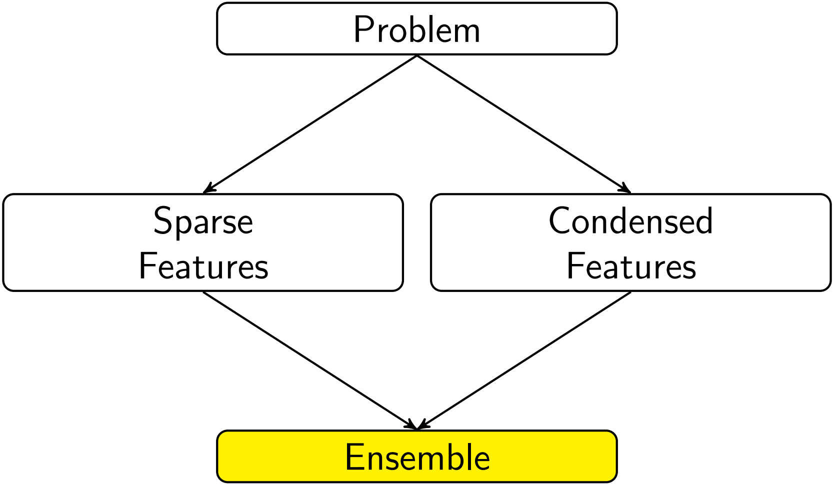
- Random Forest (Breiman, 2001) showed the best performance:

  10 decision trees with depth 7

	A89	B89
Basic sparse features	0.2895	0.2985
Best sparse features	0.2784	0.2830
Best condensed features	0.2824	0.2847
Best leader board	0.2759	0.2777

- This small feature set works well
- Due to the small feature size, a Random Forest on the training subset of a unit takes a few minutes.

Outline


Problem
Sparse Features
Condensed Features
Ensemble

Linear Regression for Ensemble

- Past competitions (e.g., Netflix Prize) showed ensemble of results from different methods often boost the performance
- We find a weight vector to linearly combine predicted probabilities from student sub-teams
- We did not use a nonlinear way because a complex ensemble may cause overfitting

Linear Regression for Ensemble (Cont'd)

- We checked linear models
  simple averaging, linear SVM, linear regression, logistic regression
- Linear regression gives best leaderboard result
  Probably because linear regression minimizes RMSE (the evaluation criterion)

Linear Regression for Ensemble (Cont'd)

- Given l testing steps and k prediction probabilities $\mathbf{p}_i \in [0,1]^l, \; i = 1, \ldots, k,$

$$\min_{\mathbf{w}} \quad \|\mathbf{y} - P\mathbf{w}\|^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2 \tag{3}$$

  λ: regularization parameter, $\mathbf{y}$: CFA vector, and $P = [\mathbf{p}_1, \ldots, \mathbf{p}_k]$.
- If $\lambda = 0$, Eq. (3) just a standard least-square problem

Linear Regression for Ensemble (Cont'd)

- In SVM or logistic regression, we may add a bias term b so

$$P\mathbf{w} \Rightarrow P\mathbf{w} + b\mathbf{1}$$

  where $\mathbf{1} = [1, \ldots, 1]^T$.
  We also replaced $\|\mathbf{w}\|^2$ with $\|\mathbf{w}\|^2 + b^2$.
- The obtained weight $\mathbf{w}$ is used to calculate $P\mathbf{w}$ for combining prediction results.
- $P\mathbf{w}$ may be out of the interval $[0, 1]$. We employ a simple truncation:

$$\min(\mathbf{1}, \max(\mathbf{0}, P\mathbf{w})), \tag{4}$$

  $\mathbf{1}$: vector with all ones; $\mathbf{0}$: vector with all zeros.

Linear Regression for Ensemble (Cont'd)

- We also explored
  Sigmoid transformation and
  Linear scaling $P\mathbf{w}$ to $[0,1]^l$,
- But results did not improve
- The analytical solution of (3) is

$$\mathbf{w} = (P^T P + \frac{\lambda}{2} I)^{-1} P^T \mathbf{y}, \tag{5}$$

  where I is the identity matrix.
- The problem is that $\mathbf{y}$ is unknown.

Estimating $\mathbf{y}$: First Approach

- Use validation data to estimate $\mathbf{w}$.
- Training set $\Rightarrow$ V and $\tilde{V}$ internally
- Student sub-teams generated two prediction results on $\tilde{V}$ and $\tilde{T}$:

$$\text{Train } V \Rightarrow \text{ Predict } \tilde{V} \text{ to obtain } \tilde{\mathbf{p}}_i,$$
$$\text{Train } T \Rightarrow \text{ Predict } \tilde{T} \text{ to obtain } \mathbf{p}_i.$$

- Let $\tilde{P}$ the matrix collecting all $\tilde{\mathbf{p}}_i$; we know true $\tilde{\mathbf{y}}$.
- In (3) using $\tilde{\mathbf{y}}$ and $\tilde{P}$ to obtain $\mathbf{w}$.
- Final prediction: we calculated $P\mathbf{w}$ and applied the truncation in (4).

Estimating $\mathbf{y}$: Second Approach

- Use leaderboard information to estimate $P^T\mathbf{y}$ in (5). We follow from Töscher and Jahrer (2009).

$$r_i \equiv \sqrt{\frac{\|\mathbf{p}_i - \mathbf{y}\|^2}{l}},$$

so

$$\mathbf{p}_i^T\mathbf{y} = \frac{\|\mathbf{p}_i\|^2 + \|\mathbf{y}\|^2 - lr_i^2}{2}. \tag{6}$$

- r_i and $\|\mathbf{y}\|$ unavailable; estimated by

$$r_i \approx \hat{r}_i \quad \text{and} \quad \|\mathbf{y}\|^2 \approx l\hat{r}_{\mathbf{0}}^2,$$

$\hat{r}_i$: RMSE on the leaderboard by submitting $\mathbf{p}_i$
$\hat{r}_{\mathbf{0}}$: RMSE by submitting the zero vector.

Ensemble Results

- We collect 19 results from 7 sub-teams
- Each result comes from training a single classifier
- To select λ, we gradually increased λ until the leaderboard result started to decline
- This procedure, conducted in the last several hours before the deadline, was not very systematic

Ensemble Results (Cont'd)

- Best A89 result: $P^T\mathbf{y}$ in (5) and using $\lambda = 10$.
  That is, second approach
- Best B89 result: using the validation set to estimate $\mathbf{w}$ and $\lambda = 0$ (no regularization)
  This means the first approach

Ensemble Results (Cont'd)

Ensemble significantly improves the results

	A89	B89	Avg.
Basic sparse features	0.2895	0.2985	0.2940
Best sparse features	0.2784	0.2830	0.2807
Best condensed features	0.2824	0.2847	0.2835
Best ensemble	0.2756	0.2780	0.2768
Best leader board	0.2759	0.2777	0.2768

- Our team ranked 2nd on the leader board
- Difference to the 1st is small; we hoped that our solution did not overfit leader board too much and might be better on the complete challenge set

Final Results

Rank	Team name	Leader board	Cup
1	National Taiwan University	0.276803	0.272952
2	Zhang and Su	0.276790	0.273692
3	BigChaos @ KDD	0.279046	0.274556
4	Zach A. Pardos	0.279695	0.276590
5	Old Dogs With New Tricks	0.281163	0.277864

- Team names used during the competition:
  Snoopy $\Rightarrow$ National Taiwan University
  BbCc $\Rightarrow$ Zhang and Su
- Cup scores generally better than leader board

Final Results (Cont'd)

Many submissions in the last week before the deadline; in particular in the last two hours

- Everyone (including ourselves) tries to achieve better leader board results
- Overfitting may be a concern

  Not very clear how serious this problem is

Leaderboard Immediately After the Deadline

Number of rows: 3419

Rows per page 100

1-100 of 3419 First | Back | Next | Last

Overall Rank	Individual/Team Name	Algebra I 2008-2009	Bridge to Algebra 2008-2009	Total Score	Date
1	BbCc	0.275893	0.277687	0.27679	2010-06-06 11:42:46
2	BbCc	0.275893	0.277687	0.27679	2010-06-06 11:44:06
3	BbCc	0.275893	0.277691	0.276792	2010-06-08 23:22:22
4	BbCc	0.275898	0.277687	0.276793	2010-06-06 11:44:14
5	BbCc	0.275893	0.277694	0.276793	2010-06-08 22:48:45
6	BbCc	0.275908	0.277687	0.276797	2010-06-08 23:37:58
7	BbCc	0.275893	0.277703	0.276798	2010-06-08 23:30:14
8	BbCc	0.275893	0.277707	0.2768	2010-06-08 23:03:36
9	BbCc	0.275893	0.277712	0.276802	2010-06-08 23:18:58
10	National Taiwan University	0.275615	0.277991	0.276803	2010-06-08 23:46:50
11	BbCc	0.275893	0.277713	0.276803	2010-06-08 22:54:56
12	BbCc	0.275893	0.277718	0.276805	2010-06-08 22:59:44
13	National Taiwan University	0.275615	0.277998	0.276806	2010-06-08 23:34:02
14	National Taiwan University	0.275615	0.277998	0.276806	2010-06-08 23:37:55
15	BbCc	0.275893	0.27772	0.276807	2010-06-08 23:11:23
16	BbCc	0.275927	0.277687	0.276807	2010-06-08 23:32:15
17	National Taiwan University	0.275617	0.277998	0.276807	2010-06-08 22:57:55
18	BbCc	0.275893	0.277728	0.276811	2010-06-08 23:16:02
19	BbCc	0.275935	0.277694	0.276815	2010-06-08 20:55:26
20	BbCc	0.275935	0.277694	0.276815	2010-06-08 22:45:33

Web Page of Final Competition Results

Log

KDD Cup 2010

Educational Data Mining Challenge

Hosted by PSLC DataShop

Prizes sponsored by Facebook, Elsevier, and IBM Research

Overview | Rules | FAQ | Downloads | Upload | Results

Winners | **Full Results**

All teams | Student teams

Final submissions of all teams with a fact sheet

Rank	Team Name	Cup Score	Leaderboard Score	Final Submission Time	Fact Sheet	Paper
1	National Taiwan University	0.272952	0.276803	2010-06-08 23:46:50		
2	Zhang and Su	0.273692	0.276790	2010-06-08 23:39:35		
3	BigChaos @ KDD	0.274556	0.279046	2010-06-07 03:48:20		
4	Zach A. Pardos	0.276590	0.279695	2010-06-08 21:31:07		
5	Old Dogs With New Tricks	0.277864	0.281163	2010-06-08 23:49:11		
6	SCUT Data Mining	0.280476	0.284624	2010-06-08 23:25:27		
7	pinta	0.284550	0.289200	2010-06-08 22:14:55		

Find: Previous Next Highlight all Match case

Outline

Diversities in Learning

We believe that one key to our ensemble's success is the diversity

- Feature diversity
- Classifier diversity

Different sub-teams try different ideas guided by their human intelligence

Diversities in Learning

We believe that one key to our ensemble's success is the diversity

- Feature diversity
- Classifier diversity

Different sub-teams try different ideas guided by their human intelligence

Our student sub-teams even have biodiversity

- Mammals: snoopy, tiger
- Birds: weka, duck
- Insects: armyants, trilobite
- Marine animals: starfish, sunfish

Conclusions

- Feature engineering and classifier ensemble seem to be useful for educational data mining
- All our team members worked very hard, but we are also a bit lucky
- We thank the organizers for organizing this interesting and fruitful competition
- We also thank National Taiwan University for providing a stimulating research environment