Applications of Scalable Machine Learning Models

MINGZHE XU
Northeastern University
April 19, 2016

1 Introduction

Machine learning is a method to enable computers to learn without explicitly being programmed.[**?**] In the late 1990s, computer scientists started to train computers to build models given the advance of digitalization and cheaper computing power. With the vast volume and increasing complexity of data, as well as accessible and affordable storage in the past decades, interests in machine learning surged unprecedentedly and applications became almost ubiquitous, which make it possible to quickly and automatically produce models that can analyze complex data and deliver faster, more accurate results. Thomas H. Davenport once wrote in The Wall Street Journal that "humans can typically create one or two good models a week; machine learning can create thousands of models a week".[**?**] Precisely. Machine learning instantaneously presents high-value predictions that can guide better decisions and smarter actions in real time without any human intervention.

1.1 The Applications of Machine Learning

The world now is swimming in machine learning products more than ever. In a variety of industries and analytical settings where access to troves of useful and reliable data is provided, machine learning is being frequently

applied to explore useful information, generate accurate prediction, and directly or indirectly increase revenues. Online retailers can recommend products closely related to customers' interests. Amazon personalized recommendations, which base on customers' order history, items in carts, items rated or liked, and products similar customers viewed and purchased, provide heavily customized browsing experience for users. This personalized marketing has generated lots of revenue. According to a survey conducted in 2015, 68.4%[**?**] of all revenue of the online retailer was from the recommendation system. Netflix also provide individualized recommendations that greatly improved user experience and more than half of the viewing is from recommendation. Similarly, YouTube's Watch Next is also an embodiment of personalized recommendation. What's more, machine learning also pops up in web search results. Search engine is the lifeblood to Google, and personalized search results as well as customized ads display are core to the click through rate of the results. They are based on the combinations of queries, ads, results, and users' browsing history. In addition to Google, Facebook's News Feed, LinkedIn's Posts also ranks in a customized order. In banking system, credit scoring, fraud detection and next-best offers are all built on machine-learning algorithms.[**?**] The recent popular computer program AlphaGo beats professional human Go players after learning from a database of around 30 million moves.[**?**] In speech and image recognition, machine-learning techniques are also widely used.

In the meantime, there are constraints. With the exceptional increase in computing power, storage capacity and network bandwidth in the past decade, datasets are growing fast in fields such as search engine, bioinformatics, IT-security, speech/image recognition, or email record, to name but a few. The growth of data size leaves computational methods the only viable way of dealing with data. However, it poses new challenges to machine learning methods. Google processes around 100 PB per day on 3 million servers; YouTube has 300PB storage and around 4 billion views per day.[**?**] The Big Data industry roughly process 2.5 quintillion bytes of data every day.[**?**] And the number is not still. Looking forward, in four years there will be 40 zetta-bytes of data. Therefore, how to make the learning process more efficient is vital to those companies. With at least 3.5 billion requests coming in everyday, what algorithms can make faster and more accurate response? What models can survive the training process on petabytes of data?

1.2 The scalability of machine learning

This explosion of data tuned widespread attention to scalability, especially learning large-scale data. Scalable machine learning involves processing paradigms, statistical analysis, algorithms for data streams, and large scale convex optimization, etc. It is an integration of system, statistics, data mining and machine learning. Scalability is one of the key concepts in Big Data. Yet far beyond Big Data, scalable or large scale was a buzzword in machine learning ever since there were large amount of data such as dealing with text document or in bioinformatics. Datasets with large number of features, samples, or even when the data can't fit into the memory, solutions are needed to enable learning and processing. Currently, there are some solutions on different levels. In memory, we can intellectually swap between memory or disk. On the algorithmic level, we can use online algorithm which can constantly learn and predict as the data flows.[**?**] Also, faster and optimized algorithms are needed. Parallel algorithms are vital to higher efficiency. When data cannot fit into memory of a single machine, distributed algorithms can tackle this problem.

In this thesis course and research work, studying scalable machine learning models and applying the concepts and theory into real world problems are the core purposes. In this semester, we've gone through several fundamental learning algorithms, as well as objective functions and models. Supervised learning problems are those with input features, such as linear regression and classification problems. Linear regression is used when the target variable to be predicted is continuous; when the prediction are discrete values such as predicting positive or negative of a certain disease, it is a classification problem. To minimize the cost functions, Least Mean Square algorithm, matrix derivatives or maximum likelihood estimate can be applied to derive the parameters for models.[**?**] Logistic regression models assumes there is one smooth linear decision boundary and it is a powerful statistical way of modeling when the dependent variable is a binomial value. It is also the most prevalent algorithm for solving the scale problem in industry given its simplicity, efficiency and robustness. Supporting Vector Machine (SVM) relies on boundary cases to build the separating curves and obtain the optimal margin classifier. SVM can handle large feature spaces and the boundary cases also enable it to handle missing data such in text classification.[**?**, **?**] Naive Bayes is a family of algorithms for training such as Gaussian naive Bayes, Bernoulli, or multinomial

naive Bayes, assuming that all values of variables are independent of other features.[**?**, **?**] Modeling with Naive Bayes can be simple and efficient while the conditions can be the constraints. Decision tree, or regression tree is derived by recursively partition the data space and fitting prediction model within each partition.[**?**, **?**, **?**, **?**] It is more intuitive and easy to interpret given the tree structure. While the complexity of the algorithm is it's main setback. In order to improve the classification rate, we can also use random forest, which applies a number of decision trees.

After laying a theoretical foundation, I am going to design and experiment large-scale modeling techniques on the dataset from KDD Cup 2012 track 2, which is on advertisement ranking. Clicking ads prediction is of great value to research given the enormous revenue it generates in industry. Predicting click through rate (CTR) is central to multi-billion dollar business in Internet marketing. Contextual advertising, display advertising and search advertising all depend heavily on an efficient learning model that can accurately predict the click through rate.[**?**, **?**, **?**, **?**, **?**, **?**] Predicting click through rate is not a brand new topic. Yet new methodologies and approaches to tackle it as well as improve the accuracy confer tremendous vitality to this subject. This thesis will focus on applying large-scale learning techniques and predicting the CTR of ads in a web search engine.

2 Data

The dataset is provided by Tencent and from the competition in track 2 of KDD Cup 2012. It includes training data, testing data and other additional supporting datasets. Training data contains149, 639,105 instances while the testing data 20,297,594 instances. They both capture the data between user and search engine, including some shared features such as UserID, DisplayUrl, KeywordID, QueryID, AdID, AdvertiserID, DescriptionID, Position and Depth. In addition, the training data also has impression and click whereas the testing data does not. For each instance, the meaning is that in a case, the user is impressed with the ad for certain number of impressions and clicked the ad number of click times. The supporting data contains token lists of query, description and keyword.

The training data contains impression and click, where impression is the number of times the ad was impressed by the user, and click is the

Table 1: Data Field Description

Field Name	**Field Description**
UserID	id for user
AdID	certain ad users impress or click
AdvertiserID	advertiser for ad, a property of ad
Depth	number of ads impressed in a session
Position	the order of an ad in the impression list
KeywordID	the keyword of ad, a property of ad it is the key of keyword token file
QueryID	id of the query zero-based integer value, key of query token file
DisplayURL	a property of the ads (hashed for anonymity)
TitleID	a property of ads, key in title token file
DescriptionID	a property of ads, key in description token file
Click	number of times the user clicked the ad
Impression	number of search sessions in which the ad was impressed by the user

number of times the impression transferred into an actual click. For every impression, the user may click or not click the ad, which can be viewed as a binary classification. In the original dataset, the click and impression are not binary-based values. Therefore a modification was made to accommodate binary classification. Each data instance was split into two with equivalent embodiment of the original CTR. For example, an instance with 4 clicks and 10 impressions will be split into 1 click with weight 4 and 0 click with weight 6. Regarding the features in the original dataset, modifications also made to have better prediction model. Each feature is represented by the corresponding value such as UserID, QueryID etc., and based these features new CTR features can be generated. For each categorical feature, we first group them and get the average click through rate for each category, then put the value back correspondingly to the original dataset as an additional column of feature.

3 Validation

The solution will be validated by area under curve (AUC) of the receiver operating characteristic (ROC) curve. ROC curve depicts the classification

performance in a two dimensional graph with false positive (FPR) rate and true positive (TPR) rate. Calculating the area under the ROC curve gives us the probability that the classifier ranks a randomly chosen positive instance higher than a randomly chosen negative instance.[**?**] Since ROC is in a unit space, AUC is a value between 0 and 1. In the coordinate, the best prediction method will yield a prediction on the upper left, (0, 1), which means that there is no false positive. This measurement is well known for ranking performance and classification performance. Though arguments exist on the coherency of AUC in measuring aggregated classifier performance, the proposed alternative was proved to be already available in the form of AUC.[**?**, **?**]

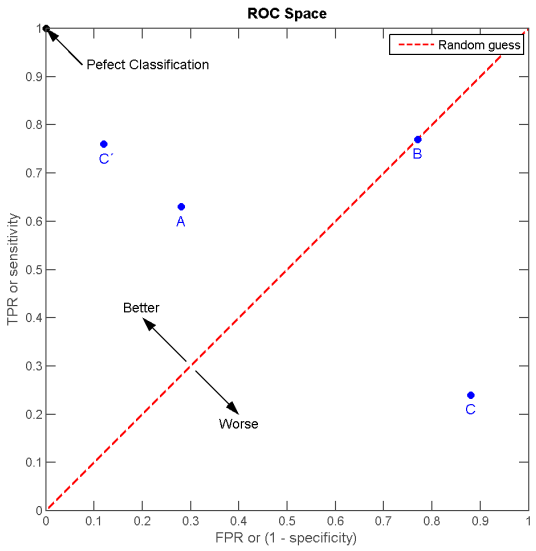


Figure. ROC Space (Wikipedia)

4 Progress

4.1 Data Preprocessing

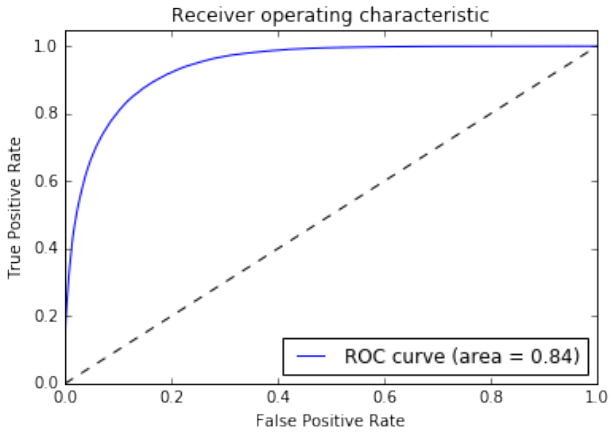
Progress has been made on both data pre-processing and some initial modeling. The original dataset contains data on user and search engine, while their values can not be directly used in models. Therefore new CTR features were generated based on the original features. Other than the given features in the original data, new features will be generated from the token

files. For features such as query, title and description, the token strings will be stemmed using Snowball stemmer.

In addition to feature modification, the original data was downsized into a 5 million instances sample in order to process faster locally. Downsizing the data into 5 million instance sample is a valid approach for this stage of proof of concept. On one side, it is faster and more efficient for model selection and feature selection where datasets need to be run multiple times so as to get the best score for features and models. On the other side, 5 million is a sufficient data sample size for initial processing of data and catch the gist of different models.

4.2 Initial Models

Logistic regression model, based on the theoretical view, is efficient for large scale learning and good for its robustness to overfitting. I first tried this model on all the features I have for now. With 5 million instances, I split the data into two parts, training and testing, 70% and 30% respectively for cross validation. Here is the preliminary result after running on my sampled data.



5 Timeline

In next semester, I am planning to learn more advanced scalable machine learning algorithms, models and techniques. In the meantime, this project will be developed into a fully-fledged thesis paper. In order to achieve that, I need to first keep taking this thesis course and study more on the related topics to hone skills in this field. Besides, here is a more detailed plan for

my thesis:

First, new features are needed to be explored from the original data. Stemming the tokens, I can then pick valuable information and construct new ads related features from description and keywords, as well as user related feature from query tokens.

Second, model selection needs to be updated given that the new features popping up. Also, with more thorough study on some advanced models, I can try ensemble modeling such as AdaBoost or Random Forest, etc.

Third, after finding the model with top AUC scores on my sample data, I will need to run the models on the full data, which contains 149,639,105 instances. Due to the size of the datasets and the constraints of my local machine, it is hard to run models locally. So, additional tasks required to mount the data to a Cloud on Amazon Web Service, Google Cloud or Azure.

References

1. Breiman, L., J. Friedman, and R. Olshen (1984). Classification and Regression trees. Wadsworth. 555 - 597
2. Chan, Philip K., and Salvatore J. Stolfo. "Toward Scalable Learning with Non-Uniform Class and Cost Distributions: A Case Study in Credit Card Fraud Detection." KDD. Vol. 1998.
3. Douglas C. Montgomery, Elizabeth A. Peck, G. Geoffrey Vining, Linear Regression Analysis, page 67-119, Edition 5, 2012
4. Drugowitsch, J. (2008). Bayesian Linear Regression. Technical report, U.Rochester.
5. D. Agarwal, B.-C. Chen, and P. Elango. Spatio-temporal models for estimating click-through rate. In Proceedings of the 18th international conference on World wide web, pages 21-30. ACM, 2009
6. Duchi, J., E. Hazan, and Y. Singer (2010), Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. http://www.jmlr.org/papers/volume12/duchi11a/duchi11a.pdf

7. Fawcett, Tom (2006); An introduction to ROC analysis, Pattern Recognition Letters, 27, 861-874
https://ccrma.stanford.edu/workshops/mir2009/references/ROCintro.pdf

8. Hand, D.J. Measuring classifier performance: a coherent alternative to the area under the ROC curve. Machine learning, 77(1): 103-123, 2009. ISSN 0885-6125.

9. Hanley, James A.; McNeil, Barbara J. (1983). "A method of comparing the areas under receiver operating characteristic curves derived from the same cases". Radiology 148 (3): 839-843.

10. Kim H, Loh WY. Classification trees with bivariate linear discriminant node models. J Computer Graphical Stat 2003, 12:512-530.

11. Kim H, Loh WY, Shih YS, Chaudhuri P. Visu- alizable and interpretable regression models with good prediction power. IIE Trans 2007, 39:565-579.

12. M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. In Proceedings of the 16th international conference on World Wide Web, pages 521-530. ACM, 2007.

13. M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. World Wide Web Conference Series, 2007

14. Rennie, J.; Shih, L.; Teevan, J.; Karger, D. (2003). Tackling the poor assumptions of Naive Bayes classifiers

15. Stuart Andrews, Ioannis Tsochantaridis and Thomas Hofmann: Support Vector Machines for Multiple-Instance Learning,
https://www.robots.ox.ac.uk/ vgg/rg/papers/andrews_etal_NIPS02.pdf

16. Thorsten Joachims: Text Categorization with Support Vector Machines: Learning with Many Relevant Features,
http://www.cs.cornell.edu/people/tj/publications/joachims_98a.pdf

17. Wei-Yin Loh: Classification and regression trees.
http://www.stat.wisc.edu/ loh/treeprogs/guide/wires11.pdf

18. O. Chapelle. Click modeling for display advertising. In AdML: 2012 ICML Workshop on Online Advertising, 2012.

19. Phil Simon (March 18, 2013). Too Big to Ignore: The Business Case for Big Data. Wiley. p. 89

20. Silver, David, et al. "Mastering the game of Go with deep neural networks and tree search." Nature 529.7587 (2016): 484-489.

21. T. Graepel, J. Q. Candela, T. Borchert, and R. Herbrich. Web-scale Bayesian click-through rate prediction for sponsored search advertising in microsofts bing search engine. 2010

22. Z. A. Zhu, W. Chen, T. Minka, C. Zhu, and Z. Chen. A Novel Click Model and its Applications to Online Advertising. http://research.microsoft.com/pubs/119092/WSDM2010.pdf

23. http://www.slideshare.net/kmstechnology/big-data-overview-2013-2014

24. http://cloudtweaks.com/2015/03/surprising-facts-and-stats-about-the-big-data-industry/

25. http://blogs.wsj.com/cio/2013/09/11/industrial-strength-analytics-with-machine-learning/

26. https://www.marketingsherpa.com/article/chart/personalized-product-recommendations