CS230

CS230 Winter 2017-2018 Project Report
Inferring Educational Preferences through Deep Learning

Zhaolei Shi
Graduate School of Education
Stanford University
zshi2@stanford.edu

Abstract

Non-linear deep learning methods may develop new insights for long-standing social science questions. In this study, I used three class of neural network models to predict students' choosing of a particular school in his or her public school application. F1 scores of lower 80% can be achieved using a Siamese-like network trained on augmented school-student paired data. Attempts to increase recall led to the adoption of a weighted cost function that increases the penalty for false negatives. Code for this project available at https://github.com/henrishi/cs230

1 Introduction

Social science methods have traditionally relied on a batch of statistical methods that have evolved separately from machine learning methods. As such, non-linear deep learning methods may develop new insights for long-standing social science questions or offer new avenues to frame social science inquiries. In this project, I explore the application of deep learning to preference predictions in the choice of schools and discuss the relevance of the results for social science research.

In this study, I used three class of neural network models to predict students' choosing of a particular school in his or her public school application. The data comes from administrative records of a large school district in the US. The networks take as input pairs of student-school characteristics. The output of the neural network is slightly different across model classes. For one class of models, the output is a softmax probability vector of length 149 (the number of schools in the choice pool). For the other two of the model classes, a binary probability of a student choosing the paired school is the output.

We have reasons to believe that preference problems are productive space for application of deep learning algorithms. There exist many potentially non-linear interactions between preferences for factors, e.g. parents may pick schools that have at least certain good test scores before preferring shorter travel distance over longer distances. Although this project investigates preference modeling in the education context, the applications of preference modeling are numerous. For example, online retailers such as Amazon or Alibaba would be able to better manage their supply chain by producing a precise model of individual user preferences and how they evolve over time.

Superior to online retail data, the school choice data for this project has a complete list of rankings among potential choices which can be used as the ground truth reference.Educational choices are also more nuanced than online purchases. Agents consider a wide range of factors and are less susceptible to strong influencing factors such as prices (in the public school context). This makes for potentially more interesting models and more useful insights.

2 Related work

There is a large body of machine literature in predicting rankings as this type of objective is common to task such as search engine queries, document retrieval, sentiment analysis, and product ratings (Cao et al., 2007; Liu and Others, 2009). A subset of the ranking literature formulates the learning problem as one of learning preferences over a set of objects (Fürnkranz and

Hüllermeier, 2010). In this formulation, the problem of learning preference functions can be formulated in one of two ways. The first formulation is object ranking, where the objective is to rank any number of objects in a complete order holding the preference fixed. The second formulation is label ranking, where a finite set of objects is held fixed and the objective is to learn how input features correspond to different preference relationships among the fixed set of objects (Fürnkranz and Hüllermeier, 2010; Kamishima et al., 2010).

One way of recovering total ranking is through training models that take pairs of input vectors and make binary classification predictions. Fürnkranz and Hüllermeier (2010) discussed how ranking problems can be characterized as generalized classification tasks. Moreover, algorithms trained on the binary classification of pairs of objects tasks has been shown to be useful in recovering the total rank order (Hüllermeier et al., 2008). Other examples include OrderSVR (Kamishima et al., 2010) and SVOR (Herbrich, 1999) where authors attempt to learn total ranking of objects by training support vector machines on pairs of objects with a binary indicator of preference.

As a part of recent advancements in deep learning, Siamese networks have been responsible for the success of many one-shot learning tasks including near human-level performance on face recognition (Taigman et al., 2014). In the original paper by Bromley et al. (1993) on Siamese networks, the authors discussed how the structure of a pair of joining vectors lends itself to naturally rank similarities between inputs. The Siamese structure may lend itself to flexibly modeling ranking relationships between agent and object pairs for social science inquiries.

This study also extends work on heterogeneous preference modeling done by economists into the space of deep learning. Recent efforts have been made by Athey et al. (2018) to estimate restaurant preferences using Bayesian latent characteristic estimations. In a similar education setting to the current study, previous work by Neilson (2013) uses structural models to estimate parental preferences for schools. A larger literature derived from consumer purchase data seeks to estimate the discrete choices of consumers (Keane, 2013).

3 Dataset and Features

3.1 General Features, Preprocessing, and Split

The data comes from administrative records of a large school district in the US. The dataset includes both student and school yearly characteristics and student choices of schools. The dataset spans the years 2005 to 2015. $110,528$ student submitted choices over this 11 year period, the choices were among 149 schools in the district.

The student characteristics include variables such as age, ethnicity, parental education, achievement levels, and residential location (GPS coordinates). The school characteristics include demographics of its students and some characteristics of its staff. It also includes student achievement information and the location of the school (GPS coordinates). In each year each student can submit up to 10 ordered choices of schools. However, many students do not utilize all of the 10 slots; the average number of choices submitted is 5.8.

Although the data has been cleaned for academic research purposes, the data had to undergo significant restructuring to feed into deep learning networks. I restructured the choice data so each record was a student-year-choice tuple. I merged student choice data with both student background and school characteristics, leaving out the records of choices without matching student characteristics. I then determined the suitable columns to use for the prediction task (values should be either measured before submitting school choices or stable characteristics that do not change frequently). I also replaced missing values with the average value of the column.

Finally, the resulting dataset after the aforementioned preprocessing step contains $486,432$ rows of student-year-choice tuples. The *training* set was created by randomly sampling 92% of this dataset. The *dev* and *test* sets were each allocated 4% of the dataset.

3.2 Defining the Types of Choices

Since in this dataset, students typically submit more than one choice of school in a particular year, multiple definitions of the what a valid choice means can be formulated. These formulations would depend on how to take into account the multiple choices that a student submits.

In the absence of a strict guideline on this issue, I defined 5 different filters for different types of choices, these filters would be used to create subsamples of the dataset for model training. *anyChoice* includes any choice the student submits. *higherChoice* includes only the schools in the upper half of a student's choice ranking. *topChoice1*, *topChoice2*, *topChoice3* include the top 1, 2, and 3 choices of each student respectively.

3.3 Negative Outcome Augmentation

So far the dataset only contains positive outcomes of schools and students matched on student choice. To create negative outcomes in this dataset, for each student-year combination, I filtered out the schools not on the list of choices for that student in that particular year and randomly sampled 3 (for some datasets also 10) non-choice schools for each choice school listed. The augmented observations receive an outcome of 0 indicating non-choice, while the original observations receive an outcome of 1 indicating a valid choice. The resulting dataset contains $1,945,728$ observations, 4 times the original number of observations. The 3-augmented datasets will hereon be referred to by *3 Aug* and the 10-augmented datasets, *10 Aug*.

4 Methods

To answer positive inquiries in social science, the models need to take into account the information embedded in student and school characteristics to make preference predictions. As such, the following three classes of models were used to take advantage of student and school feature vectors. In each of these models, the feed-forward mechanism works as follows. Let $a^{[l]}$ be the activation of the lth hidden layer, we have $a^{[l]} = g^{[l]}(z^{[l]})$ where $z^{[l]} = w^{[l]T}a^{[l-1]} + b^{[l]}$ and $g(.)^{[l]}$ is the activation of the lth layer. $a^{[0]} = x$ is the input feature vector. Normalization was performed on each input vector before they were fed into the networks. These models draw inspiration from various papers mentioned in related work:

1. *Categorical* is a simple baseline model using feed-forward network to accomplish multi-class classification (among the 149 candidate schools). The output of the model is a softmax function where $\sigma(z)_j = \frac{e^{z_j}}{\sum_k^K e^{z_k}} \quad for \quad j = 1, 2, \ldots, K$ where K is the number of classes. This model only takes as input the student feature vectors and does not require the use of the augmented negative outcomes.
2. *Stacked* is a feed forward network taking in a joined vector of student and school features to make a binary classification. In this case the output of the model is a sigmoid function where $\sigma(z) = \frac{1}{1+e^{-z_k}}$. This model is inspired by the pairwise classification-based approaches to retrieve utility function. This models used both positive and negative outcomes.
3. *Siamese* is a Siamese-like network where two networks are created separately for student and school features. Each network feeds into two hidden layers of equal size. The absolute value of the difference of these hidden layers is fed into a dense layer to model binary probability. The hidden feature layer can be expressed as $f(x_{stu})$ and $f'(x_{sch})$ for students and school separately. The z for the output layers is therefore $z = \sum_j w_j|f(x_{stu})_j - f'(x_{sch})_j| + b$ summed over the length of the vector. z is then fed into the same output function as *stacked*. This model is inspired by the pairwise approaches and the Siamese network's natural capacity to model distance and orderings. This models used both positive and negative outcomes.

For all three class of models, the cross entropy loss is used. Cross entropy for the multiple-class cases for an observation i can be expressed as $\mathcal{L}(\hat{y}^{(i)}, y^{(i)}) = -\sum_j y_j^{(i)} \log \hat{y}_j^{(i)}$ where j is the index of the softmax vector. In the binary prediction case, this becomes $\mathcal{L}(\hat{y}^{(i)}, y^{(i)}) = -[y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})]$.

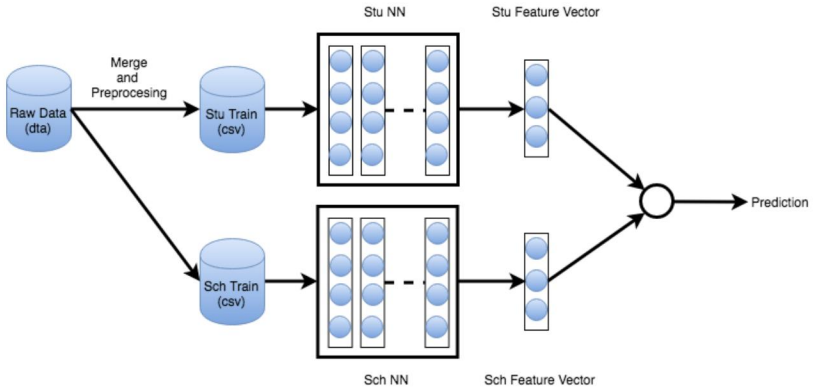

Figure 1: Siamese Network Structure

5 Experiments/Results/Discussion

All models in this study are implemented with tensorflow (Abadi et al., 2016). Structural starter code was provided by the CS 230 teaching team at Stanford.

5.1 Model Space Search

The first task is to compare the performance of different models classes and choose one to explore in depth. I systematically compared the three performance of three model classes by sampling a range of models from each class with different layers and nodes (details in Hyper Parameter Section). The best performance out of candidates of the categorical model class rests at 28.7%. For the other two model classes, I use the F1 scores, the harmonic average of precision and recall, as the main evaluating metric. The best performing Stacked and Siamese model class seems to be doing equally well on the 10 Aug data. However, more models in the Siamese class ranked among the top performers for this data. The Siamese structure also enjoys natural mapping to a "distance" metric between student and school feature vectors, which would be useful in social policy applications. Thus, from here on, the Siamese model is chosen as the main model class of exploration.

Table 1: Baseline Performance of Different Models Classes

Model Class	Accuracy	Precision	Recall	F1 Score	Input Data
Categorical	0.287	-	-	-	Top Choice (Student Data Only, No Augmentation)
Stacked	0.943	0.726	0.597	0.678	Top Choice Student and School Data (10 Augmented)
Siamese	0.942	0.689	0.667	0.678	Top Choice Student and School Data (10 Augmented)

5.2 Hyper Parameter Space

Through initial trial and error attempts at training the data interactively, I was able to develop some intuition about the ranges of hyper parameter search. The plausible optimal learning rate range seems to be $[10^{-4}, 10^{-2}]$. In addition, a more significant hyper parameter seems to be the number of layers of the neural network and how many nodes exists in each layer. The plausible range of layers seems to be $[1, 5]$ and the number of nodes of each layer between $[20, 1000]$. Other hyper parameters also include dropout rate, feature vector size, and the weights of the cost function (details in next section). In total, around 400 model-data combinations were sampled, trained, and evaluated.

To better engage this multi-dimensional sample space, I implemented a random sampler to create models with different sets of hyper parameters. Certain hyper parameters such as node numbers and learning rate were randomly sampled on a log scale, while number of layers, dropout rate etc. were sampled in their original value scale. The random sampling of these hyper parameters allows me to more efficiently explore the sample space.

Notably, in my sampling process, I favored network structures with larger layers closer to the input and smaller layers closer to the output. This is done for two reasons, 1) there is reason to believe lower level interactions between input features of the data can be more useful than higher level features for this training task, 2) the Siamese network necessarily condenses into a small feature vector at the very end. To favor this type of structure in the model sampling process, each layer is ordered based on the size of its nodes. However, a "switch" parameter is implemented so that any two layers settings has a random probability of switching their order. This allows for exploration of networks ordered in a different way.

5.3 Effect of Different Cost Functions and Dropout Rate

Table 2: Dropout and Weighted Cost (Top Choice 10 Aug)

Model Specification	Accuracy	Precision	Recall	F1 Score
Without Dropout Nor Label Weights	0.945	0.730	0.634	0.679
Dropout Rate = 0.3	0.945	0.744	0.611	0.671
Positive Label Weight = 2	0.939	0.649	0.724	0.684

One problem encountered in earlier iterations of the experimentation was that due to the larger number of negative-outcome samples relative to the positive-outcome samples, the models were producing low recall rates. My attempt to solve this problem was to implement a larger weight for positive labels in the loss function. This weight seems to produce the best results when set between $[1, 2]$. Table 2. illustrates the effect of weighting the loss this way on a particular network with the top choice 10 Aug data.

Another problem that only existed for larger models was the eventual over-fitting of the training data. This was more limited than the recall rate problems. I implemented a dropout layer after the input layer and experimented with the dropout rate as a hyper parameter to be sampled. Table 2 shows the result of implementing a large dropout rate of 0.3, for this particular network, this seemed to have little effect on the general performance of the model.

5.4 Best Siamese Performance

Table 3. presents the best performing Siamese models for each dataset. The F1 score for 3 Aug data seems to hover consistently at slightly over 0.8 regardless of the different definitions of how choice outcome was defined. For a larger augmentation of 10, the best Siamese performance slightly breaks 0.7. We note that the best performing model is also larger for this model than optimal models for 3 Aug data. This suggest that a larger network structure may work better for cases where the negative outcomes occupies a larger fraction of the sample.

Table 3: Best Siamese Model Performance

Data Set	Accuracy	Precision	Recall	F1 Score	Network Structure (Nodes Per Layer)	Feature Vector Size
Top Choice (10 Aug)	0.944	0.677	0.733	0.704	Stu: [875-263-454-152-576-20] Sch: [292-411-20]	20
Top Choice (3 Aug)	0.907	0.779	0.893	0.832	Stu: [123-50-432-20] Sch: [251-175-20]	20
Top 2 Choice (3 Aug)	0.910	0.812	0.848	0.829	Stu: [143-68-110-63-20] Sch: [59-326-818-20]	20
Top 3 Choice (3 Aug)	0.907	0.791	0.862	0.825	Stu: [519-268-72-164-20] Sch: [720-656-20]	20
Any Choice (3 Aug)	0.904	0.788	0.849	0.817	Stu: [143-68-110-63-20] Sch: [59-326-818-20]	20
Higher Choice (3 Aug)	0.906	0.775	0.882	0.825	Stu: [121-133-20] Sch: [251-130-20]	20

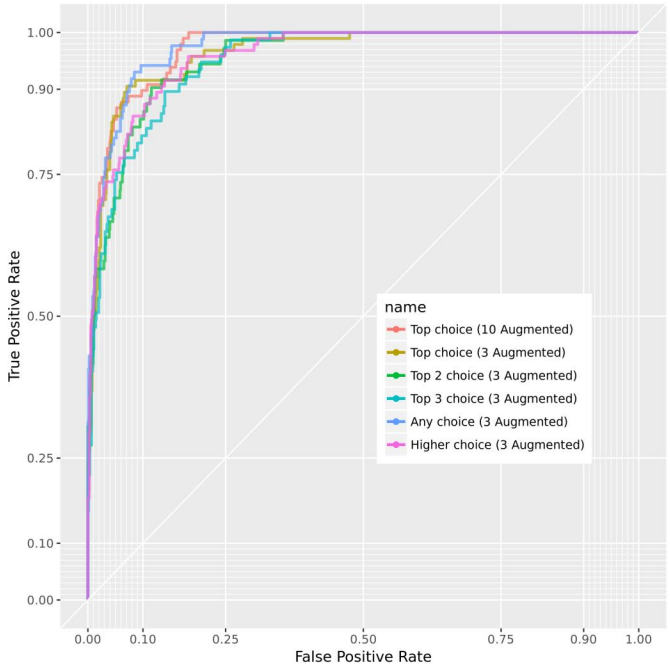

Figure 2: ROC Curves for The Best Siamese Models by Dataset

5.5 Discussion and Future Work

The Siamese model performed decently on the student-school paired data. The next step of future work on this project is to investigate the performance of these models on the recovery of the complete preference order (not only binary outcomes) of students. Another extension is the inclusion of temporal dependence for multiple choices made by the same student through time. Finally, a counter-factual analysis of preference changes due to exogenous variations in choice set would elevate the value of these models for social scientists and policy makers.

6 Contributions

This project was a sole-contributor project.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I. J., Harp, A., Irving, G., Isard, M., Jia, Y., Józefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D. G., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P. A., Vanhoucke, V., Vasudevan, V., Viégas, F. B., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, abs/1603.04467.

Athey, S., Blei, D., Donnelly, R., Ruiz, F., and Schmidt, T. (2018). Estimating heterogeneous consumer preferences for restaurants and travel time using mobile location data. *arXiv preprint arXiv:1801.07826*.

Bromley, J., Guyon, I., Lecun, Y., Sackinger, E., and Shah, R. (1993). Signature verification using a siamese time delay neural network. In Cowan, J. and Tesauro, G., editors, *Advances in neural information processing systems (NIPS 1993)*, volume 6. Morgan Kaufmann.

Cao, Z., Qin, T., Liu, T.-Y., Tsai, M.-F., and Li, H. (2007). Learning to rank: From pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 129–136, New York, NY, USA. ACM.

Fürnkranz, J. and Hüllermeier, E. (2010). *Preference Learning*, pages 789–795. Springer US, Boston, MA.

Herbrich, R. (1999). Support vector learning for ordinal regression. *IET Conference Proceedings*, pages 97–102(5).

Hüllermeier, E., Fürnkranz, J., Cheng, W., and Brinker, K. (2008). Label ranking by learning pairwise preferences. *Artificial Intelligence*, 172(16):1897 – 1916.

Kamishima, T., Kazawa, H., and Akaho, S. (2010). A survey and empirical comparison of object ranking methods. In *Preference learning*, pages 181–201. Springer.

Keane, M. (2013). Panel data discrete choice models of consumer demand. Economics Papers 2013-W08, Economics Group, Nuffield College, University of Oxford.

Liu, T.-Y. and Others (2009). Learning to rank for information retrieval. *Foundations and Trends®in Information Retrieval*, 3(3):225–331.

Neilson, C. (2013). Targeted vouchers, competition among schools, and the academic achievement of poor students.

Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708.