Semi-Supervised Learning for Predicting GMAT Scores

Yi-Fu Wu

Abstract

We apply a method of applying semi-supervised learning to data from an online test preparation site in order to predict students' GMAT scores. We leverage a large amount of unlabeled data by using a variational autoencoder to extract feature embeddings for each student. We compare these feature embeddings with hand engineered features on a linear regression model and show that the feature embeddings perform comparably. We also show that combining the feature embeddings with the hand engineered features increases the performance of the model with hand engineered features alone.

Dataset

We use a dataset from TAL Education Group, an online education company that provides online test prep for various exams including the GMAT. The original dataset consists of 8,681 questions with category labels (verbal, quantitative, integrated reasoning); 90,831 students; 16,002,324 student-question interactions consisting of a student, a question, a time the question was attempted and whether or not the student got the question correct; and 458 students labeled with self-reported GMAT scores. Out of the 458 students with self reported GMAT scores, 372 used the test prep system before the date of their reported exam. Out of these 372 students, we consider the 354 students who attempted at least 50 questions. Similarly, out of the 90,831 overall students, we only consider the 19,841 students that have completed at least 50 questions. The overall number of questions that these students attempted is 7808.

References

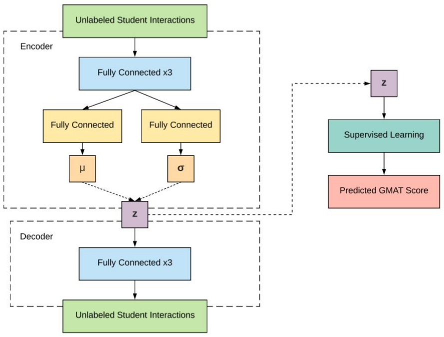
[1] Diederik P. Kingma, Danilo Jimenez Rezende, Shakir Mohamed, and Max Welling.
Semi-supervised learning with deep generative models.
In *NIPS*, 2014.

[2] Diederik P. Kingma and Max Welling.
Auto-encoding variational bayes.
CoRR, abs/1312.6114, 2014.

[3] Severin Klingler, Rafael Wampfler, Tanja Käser, Barbara Solenthaler, and Markus H. Gross.
Efficient feature embeddings for student classification with variational auto-encoders.
In *EDM*, 2017.

[4] Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther.
Ladder variational autoencoders.
In *NIPS*, 2016.

Model



We use a variational autoencoder (VAE) to create feature embeddings for the students in our dataset with the following objective:

$$\mathbf{E}_{q_\phi(z|x)}[\log P_\theta(x|z)] - \beta D_{KL}[q_\phi(z|x)||P(z)]$$

First, we use the unlabeled student-question interactions as input to a VAE to generate feature embeddings z. The VAE uses three fully connected layers as both the encoder $q_\phi(z|x)$ and the decoder $P_\theta(x|z)$. The encoder uses another set of fully connected layers to generate $\mu_\phi(x)$ and $\sigma_\phi(x)$. We use these as parameters to a normal distribution and sample using the reparameterization trick to generate our embeddings z.

Next, we use these feature embeddings for supervised learning on our labeled dataset to predict GMAT scores. We run experiments that use the feature embeddings alone to make the predictions as well combining the embeddings with our original hand engineered features.

Results

Random Seed	Features	RMSE	MAE	R^2
1	Hand engineered	68.5044	52.4286	0.3351
1	VAE	75.4983	56.0000	0.1924
1	VAE + Hand engineered	68.4418	53.2857	0.3363
2	Hand engineered	49.6560	39.4286	0.2993
2	VAE	52.3450	40.2857	0.2214
2	VAE + Hand engineered	49.5263	38.4286	0.3030
3	Hand engineered	54.8895	41.5714	0.4161
3	VAE	68.2352	50.2198	0.2231
3	VAE + Hand engineered	51.6582	39.4286	0.4828
4	Hand engineered	67.1884	46.2857	0.2690
4	VAE	65.9329	47.8571	0.2960
4	VAE + Hand engineered	62.1174	42.7143	0.3751
5	Hand engineered	74.3928	53.4286	0.1935
5	VAE	70.0306	51.2857	0.2853
5	VAE + Hand engineered	62.1174	42.7143	0.3072

Discussion

Our overall R^2 scores are low, indicating that both the hand engineered features and the VAE generated features may not be the best model for predicting GMAT scores in the dataset. Our VAE generated features generally performs comparably to the hand engineered features, although there is a bit of variance. However, combining the hand engineered features with the VAE features seems to increase the R^2 across the board. This indicates that we can indeed leverage the large amount of unlabeled data in this dataset to improve the performance of our supervised learning model. Although most of the improvements of combining the features are modest, the random seed 4 and random seed 5 runs show that significant improvements in R^2 is possible by leveraging the VAE generated data.

Future Work

- Explore using the extracted feature embeddings on other regression algorithms such as decision trees, SVM regression, or gradient boosting methods
- Experiment with different VAE architectures such as ones that include convolution or recurrent layers