Techniques for Optimizing Information Exchange in Educational Settings: Stack Exchange as Case Study

Kevin Chen (kchen42@stanford.edu), Andrew Slottje (slottje@stanford.edu), Nurbek Tazhimbetov (nurbek@stanford.edu)

Institute for Computational and Mathematical Engineering, Stanford University

Objectives

Learn the way to efficiently write posts on Stack Exchange in response to users' questions in such a way as to best convey information, and thereby become the accepted answer.

Data

Our data were obtained from the Stack Exchange data explorer tool compilation on Kaggle. Included features comprise:

- Questions dataset: text of R questions on Stack Exchange with ID, score, post date, title and answer bodies
- Answers dataset with text, ID, associated question, and indicator variable for accepted answers (response variable)

Data Preprocessing

We provided additional features on top of the ones already in the dataset for analysis.

- Standardization: we standardized answer score and body length by score and length of the associated question.
- Transformation: absolute time $t \to \Delta t$, the elapsed time between question and answer.
- Data quality control: we had to drop some observations. For example, we dropped questions with score < 5, as they did not evince community benefit and so did not contribute to valuable prediction.

Temporal Features

We hypothesized that new answers on old questions may not be as likely to have very many upvotes. Because visual inspection shows that the score decreases with time, we divide the data in 5000 quantiles and find the maximum score associated to each quantile. This provides an upper bound on the way increased time elapsed in an answer decreases its associated utility. We recognize a loglinear relationship and OLS regression confirms this effect, visible at right.

Naive Bayes Analysis

We performed naive Bayes analysis, extracting into a dictionary the morphological stems of the words contained in the answer data. We managed the dataset to control the length of included words, inclusion of numerary values, etc. We additionally added some HTML tags manually, as the stem extraction removed these. We then trained a Bernoulli naive Bayes model to classify successful answers. Our accuracy was 62%. Top predictors included the following (in decreasing order of rank):

- HTML tags for paragraph separation
- HTML tags for code and fixed-width formatting (i.e., code)
- "You"
- Outside links ("href" from HTML link tags, "http")
- Visual support: "img," "imgur," "plot," "png"

Model Selection

We implement multiple models including ℓ_2-regularized logistic regression, GDA, and random forest. We use "accepted answer" as the response variable and use as features the score from naive Bayes and time elapsed from question to answer. We do not see any increase in predictive power with other features. When we implement the estimated algorithms on the test set, we find similar performance among the top models:

Model	Accuracy
Random Forest	66.48
Logistic Ridge	65.74
GDA	65.64

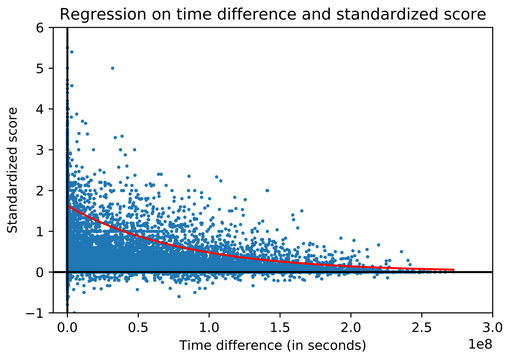

Figure 1: Regression justifying use of $\alpha e^{-\beta \Delta t}$ in feature set

Analysis of Results

Because our data were relatively tractable to the implementation of naive Bayes, we had a very successful baseline to build from. The most valuable feature besides the score from naive Bayes for improving model performance was time elapsed since question post. Improvements beyond naive Bayes were marginal, as evidenced when plotting ROC, which shows universally high specificity and low sensitivity. The naive Bayes score therefore provides the bulk of predictive power, with only modest improvement in accuracy from inclusion of quantitative predictors and limited ability to improve on this model regardless of the modeling algorithm.

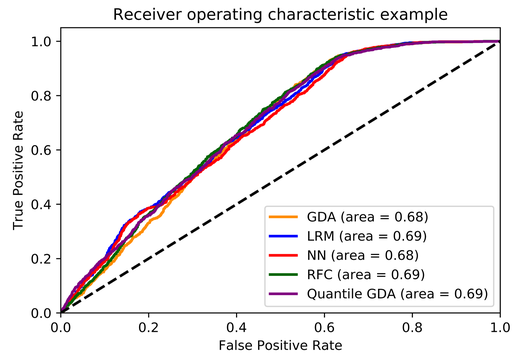

Figure 2: ROC curves for selected models

Nonetheless, we can extrapolate several successful strategies from the naive Bayes results, which include clear separation of guidance into paragraphs, providing concrete examples of good code rather than abstraction, and visual illustration, which is another means of providing examples. Time elapsed also provides a good predictor, and this indicates that prompt response is another factor that should be prioritized in online communication and in Stack Exchange in particular.

Discussion

With the increasing importance of online interactivity in education, it is more important than ever to pioneer effective strategies for communicating educatively online. [1, 2] Our work provides some insight into the best way to engineer strategies for this kind of communication on Stack Exchange. There exist at least two important directions for future research:

- Use of more advanced NLP methods to predict successful answers based on blocks of phrase as well as grammatical structure.
- Further calibration of the model we have developed using these measures to supplement the naive Bayes scores.

Model Illustration: Linear GDA

We provide some of the graphs we used to visualize the effect of predicting the response using linear Gaussian discriminant analysis on the data.

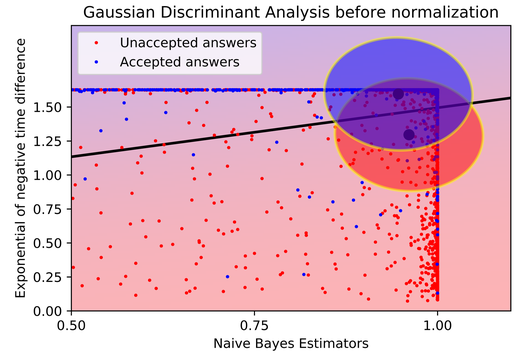

Figure 3: Results with GDA analysis

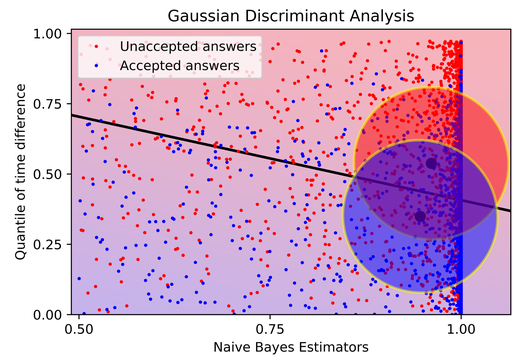

Figure 4: Results with GDA analysis under feature normalization

References

[1] I. Elaine Allen and Jeff Seaman.
Changing course: Ten years of tracking online education in the united states.
Babson Survey Research Group, 2013.

[2] Felizian Kuhbeck, Stefan Engelhardt, and Antonio Sarikas.
Onlineted.com - a novel web-based audience response system for higher education. a pilot study to evaluate user acceptance.
GMS Zeitschrift fur Medizinische Ausbildung, 31(1):Doc5, 2014.

[3] Linda Harasim.
Shift happens: online education as a new paradigm in learning.
The Internet and Higher Education, 3(2):41–61, 2000.