Ranking Course Reviews: A Classification Approach

Griffin Tarpenning, Bhagirath Mehta,* Leif Jurvetson**

**CS221, Computer Science, Stanford University*

Stanford Computer Science

Abstract

User reviews are the corner stone of countless recommendation systems, from Amazon to Netflix, and yet many reviews are meaningless or minimally informative. We explore Stanford course reviews with the goal of removing general or non-specific reviews.

Employing a novel approach to review ranking, we frame a review's utility as its specificity to a given department. Stanford's 5,000+ classes would prove impossible for rote classification; however, using a class' department as a proxy for topicality, we are able to achieve significant results.

Our LSTM model reaches over 55.8% accuracy for classifying reviews for the top 10 most reviewed departments and 32.3% on the top 100. However, perhaps more important than accuracy is the confidence of each prediction, which can be used to determine whether a given review is a general or specific claim. For example, "This course is great" is hard to assign to a department, while "I loved learning about sauropsids" can confidently be assigned to BIO.

Our results show that department classification is a valid, if simplistic, way of purging non-course-specific, and therefore, less informative reviews.

Stanford University

Infrastructure—Data Gathering

The Data

Stanford course reviews range from eloquent to livid to monosyllabic. Below are examples of reviews from CS221 (Spring 2019). Enjoy!

> Enjoy!
> (2018/2019, Spring)

> Although I did extremely well on all of the assignments, I still feel like I don't understand many of the topics covered in this course, which is rather unfortunate.
> (2018/2019, Spring)

> A great introduction to AI. Start on the project early and be prepared to spend quite a bit of time on the Problem Sets.
> (2018/2019, Spring)

> Expect no weekends free
> (2018/2019, Spring)

Gathering

Due to sensitive review data and the internal workings of Stanford systems, we manually scraped the web for our review data. Stanford course reviews are stored by course/section leading to ~10,000 pages (8.4 GB) that required scraping. The following methods were employed to capture the review text and course metadata:

- multithreaded bash/wget script for downloading the full contents of each page
- BeautifulSoup for html digestion with a javascript renderer
- ExploreCourses API

Processing

Text processing methods were crucial to model success; the order of manipulations was as follows:

- special character removal
- punctuation replacement
- whitespace cleaning
- tokenization
- conversion to embedding with 20k unique word cap

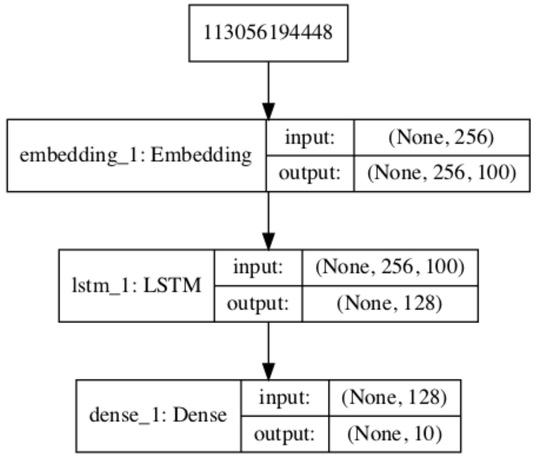
Characteristics

- 46985 reviews in top 10 most reviewed depts.
- 119559 reviews in top 100 most reviewed depts.
- avg. review length: 23.1 words
- avg. word length: 5.75 characters

Model

Overview

The experimental model used was a single direction Long Short Term Memory (LSTM) neural network with pre-trained GloVe embeddings. The structure is graphically outlined below for classifying reviews in the top 100 most reviewed departments.



Method/Reasoning

- Embedding layer: Looks up each token in input; outputs unique vector. Large matrix multiplication with 1-hot input for each token.
- Long short-term memory layer: State is maintained in the form of 128 internal state neurons. Identical cell is applied to each token of the input from left to right.
- Dense layer: Final LSTM output after final embedded token is fed into dense layer. Every neuron of the final has a connection to every neuron of the last LSTM cell.
- Standard softmax activation function applied at end is what guarantees that all of our outputs sum up to 1
- Utilized 90/10 training/test split

Results

Baseline Results

Model	10	100
Human Oracle	49%	28%
Naive Bayes bigram:	43.7%	18.24%
Naive Bayes trigram:	33.83%	15.41%

Experimental Results

Model	10	100
LSTM	55.78%	32.30%

Error Analysis

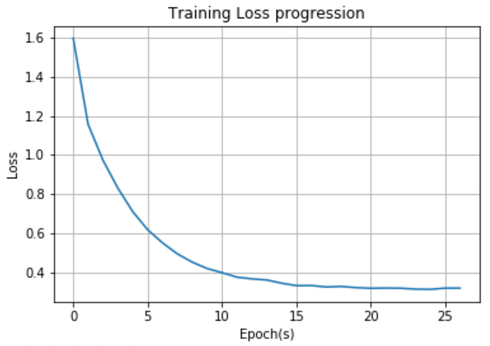
Looking at missed predictions is imperative to the utility of our model, as missed predictions occur when reviews lack department specificity. Below are random misses from the top 10 model.

ID	Pred	Actual	Text
1	HIST	ME	Great course Great professor
2	CEE	POLISCI	Organized on top of responding to email
3	CS	ME	Willingness to go beyond the scope of the course to explain concepts
4	CS	ME	awesome
5	CS	EDUC	I absolutely loved this class and highly recommend it
6	HUMBIO	ME	Ability to care – she is so caring and can calm you down if you re a hot mess Wow such a great human

Common Themes:

- general descriptions of instruction
- method description vs. content

Results — Continued



Discussion — Future Steps

- Given what our assumption overlooked (helpful reviews that described a teacher's method, if not the class content), we would use a feature selection algorithm; one that utilized the score from the LSTM, and others that looked for keywords (similar to sentiment analysis). By utilizing multiple criteria, we could account for the fact that descriptive reviews can contain information on both the content of the material and methodology of teaching used in the course.
- To expand on our use case, we can train on more data from previous years, and test on future years. We would also like to segment not just by department, but also by level (e.g. 100 vs 200 classes).