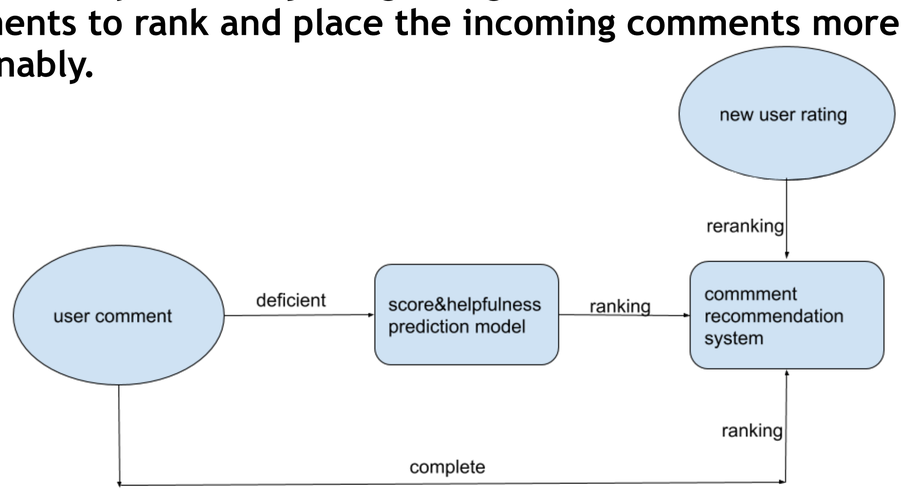
Google ML Winter Camp
谷歌机器学习应用冬令营

Comment Grading for Recommendation

Hengyi Wang, Yaofeng Sun, Jiaqi Xi

• Introduction

Real world product comments are ranked according to product scores from consumers and helpfulness evaluation from viewers. However, when these grades are deficient, the recommendation system suffers from Cold Start Problem. Thus, we established an interactive system for joint grading and recommendation of user comments to rank and place the incoming comments more reasonably.



Helpfulness Measurement

$$h = sigmoid(Helpfulness\_Denominator) \cdot \frac{Helpfulness\_Numerator}{Helpfulness\_Denominator}$$

Model error $\epsilon = E_{x\sim D}(y_{pred} - y_{true})$

System error $\Delta = rank_{pred} - rank_{true}$

19	quaker oatm	1	3		19	quaker oatm	1	3
20	a back i orde	1	3		20	a back i orde	1	3
21	i order stuff	1	3		21	i order stuff	1	3
22	i though i w	5	2		22	i though i w	5	2
23	i pick box go	4	2		23	had crave so	5	2
24	of 34 k cup	4	2		24	i pick box go	4	2
25	i deserv get	1	2		25	of 34 k cup	4	2
26	had crave so	1	2		26	i deserv get	1	2
27	do purchas	4	1		27	do purchas	4	1
28	give reason	4	1		28	give reason	4	1
29	even sale pr	1	1		29	even sale pr	1	1
30	to make edi	1	1		30	to make edi	1	1

Baseline System **Ideal System**

• Grade Prediction

The Baseline model is a neural classifier based on LSTM. We then trained a Multimodal Multitask Classification model via Fine Tuning of BERT on Amazon Fine Food Reviews dataset.

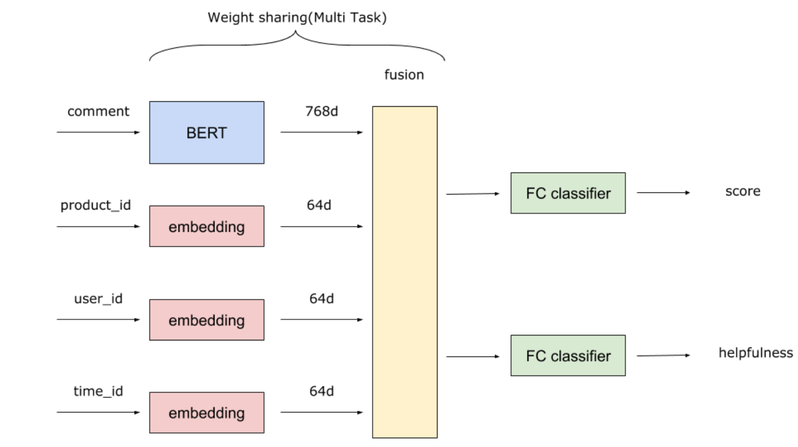
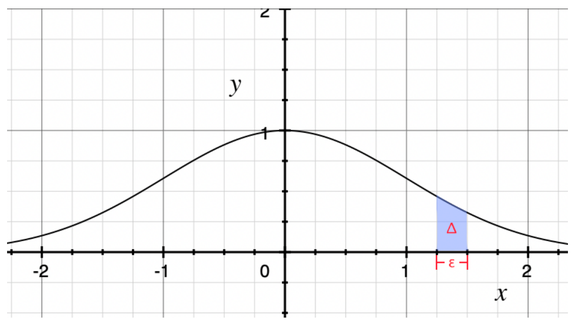


Table 1: Model Comparison

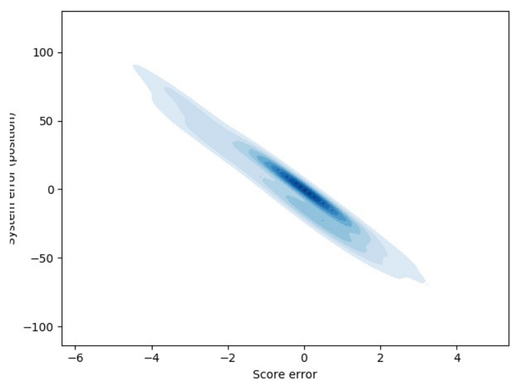
Model	Score (Acc.)	Helpfulness (Acc.)
SingleLSTM	47.4%	33.3%
SingleBERT	51.3 %	35.9 %
BERT MulMod	51.5 %	35.9 %
BERT MulMod MulTask	50.7 %	34.4 %

• Recommendation System

Generally, comments are sorted according to their score and helpfulness. When score or helpfulness is deficient, the neural model will predict labels for the two grades in advance. Our system weighs between the error of model prediction and user rating, to enable helpfulness to converge to ground truth quickly and stably.

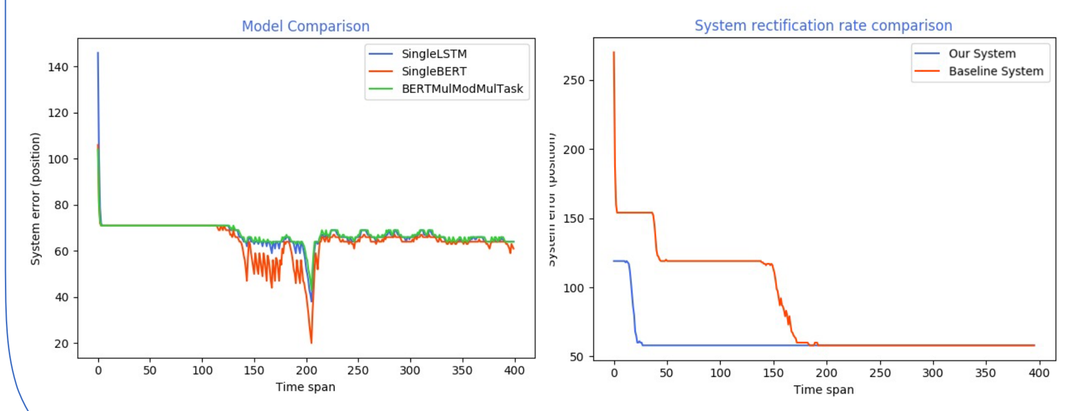


System error induced by model error



$$h_0 = h_{model}$$
$$for\ t = 1\ to\ T$$
$$x_{t+1} = x_t + z_t,\ where\ z_t \sim B(1, h_{true})$$
$$y_{t+1} = y_t + 1\ ,\ h_{t+1} = \frac{x_{t+1}}{y_{t+1}}$$
$$h_{t+1} - h_t = \frac{x_{t+1}y_t - x_t y_{t+1}}{y_t y_{t+1}} = \frac{z_t + h_t}{y_{t+1}}$$

When model error is relatively small, Cold Start can be alleviated, while the helpfulness takes more time steps to converge.



$$h_{t+1} = \frac{x_{t+1}}{y_{t+1}} = \frac{y_0 h_0 + \sum_{i=0}^{t} z_t}{y_0 + t}$$
$$\epsilon_n = \sum_{t=1}^{n} (z_t - h_{true}) \sim N(0, \frac{1}{n})$$
$$Let\ \epsilon = \frac{1}{\sqrt{y_0}} \quad y_0 = \frac{1}{\epsilon^2}$$

During initialization, the virtual time step is allocated, so as to balance the error introduced by model prediction and user rating.

投 票 区 域