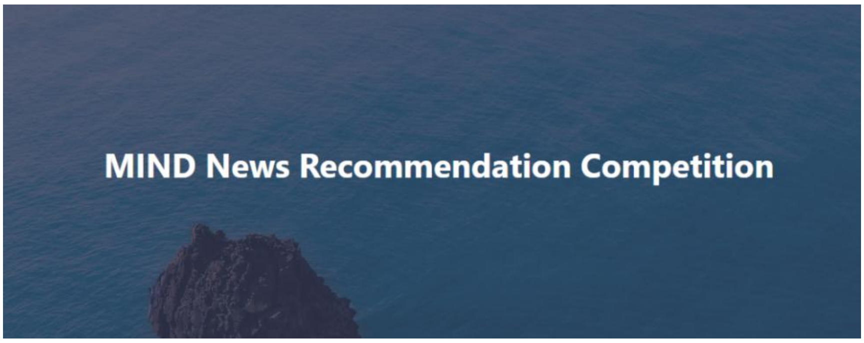


MIND News Recommendation Technical Report

Author : Zhenghong Yang
Team : YangZhenghong

Device :

CPU:I7 9700K GPU: RTX 2070 Memory : 56GB

Background

Online news services such as Microsoft News have gained huge popularity for online news reading. However, since massive news articles are published everyday, users of online news services are facing heavy information overload. Therefore, news recommendation is an important technique for personalized news services to improve the reading experience of users and alleviate information overload.

However, news recommendation is a challenging task. First, news articles on news websites emerge and update very quickly. Many new articles are posted continuously, and existing news articles will disappear after a short period of time. Thus, there is a severe cold-start problem in news recommendation. Second, news articles usually contain rich textual information such as title and body. It is very important to understand news content from their texts using NLP techniques. Third, there is no explicit rating of news articles posted by users in news platforms. Thus, in news recommendation we need to model users' interests from their browsing and click behaviors. However, user interests are usually diverse and dynamic, which poses significant challenges to user modeling algorithms. Thus, further researches are highly needed to tackle the various challenges in news recommendation.

Task

The task in this competition is described as follows. Given the news browsing history [n1, n2,..., nP] of a user u and a set of candidate news [c1,c2,...,cM] in an impression log, the goal is to rank these candidate news articles according to the personal interest of this user. In this process, news articles can be modeled by their content, and users' interests can be modeled by their news browsing history. Then, the model predicts the click scores of candidate news based on the relevance between candidate news and user interests. Finally, the candidate news articles in each impression are ranked by their click scores. The ranking results will be compared with the real user click labels to measure the ranking quality via several metrics including AUC, MRR and nDCG@K (see Evaluation tab).

Evaluation

Systems are evaluated using several standard evaluation metrics in the recommendation field, including: area under the ROC curve (AUC), mean reciprocal rank (MRR), and normalized discounted cumulative gain for K shown recommendations (nDCG@K). The final result is the average of these metrics on all impression logs. The primary metric for submission ranking is AUC.

Exploratory Data Analysis

The MIND dataset contains 1,000,000 users and 161,013 news articles. There are 2,186,683 samples in the training set, 365,200 samples in the validation set, and 2,341,619 samples in the test set, which can empower the training of data-intensive news recommendation models.
As fig.1 shown, different news' category are different from each other.

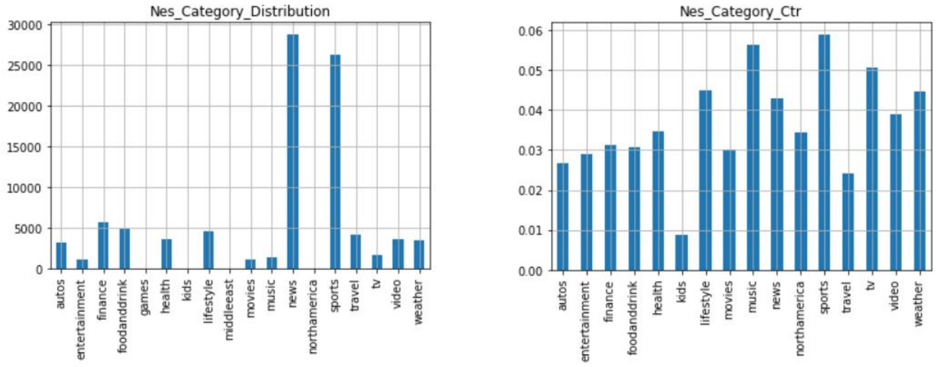


Figure 1: (a) The bar plot of news' category count (b) The bar plot of news' category ctr

Most news' lifetime is less than 3 days. Fig.2 shows that the first 10 hours is the most important time to a news.

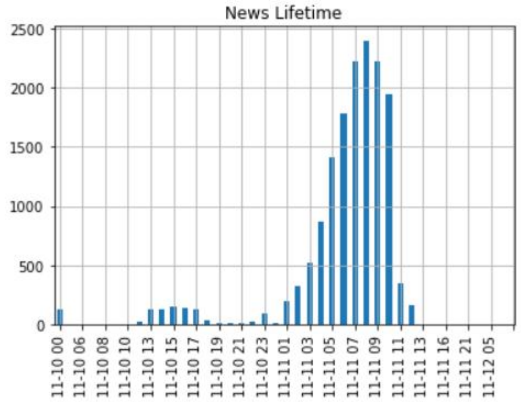


Figure 2: News exposure lifetime plot

Fig.3 shows the distribution of news' title length and entity count.

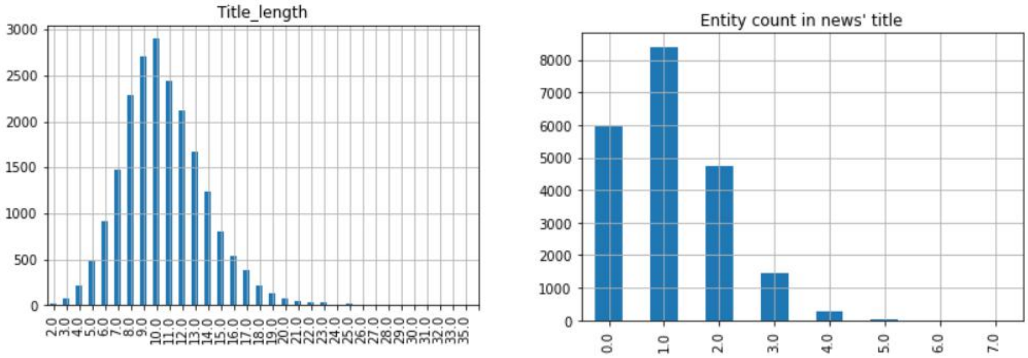
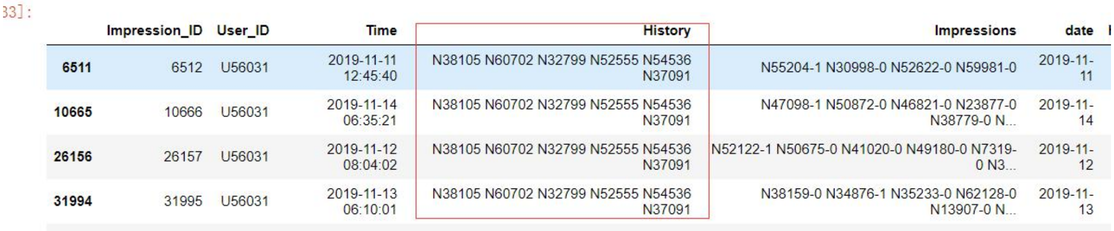


Figure 3: (a)Title length distribution (b)Title's entity count distribution

As we knew the click history is constructed by the first four weeks click behaviors. But we need to predict one week's data in test set.So it will lead a big gap in different days. Such as on the 1st day we don't miss any click behaviors. But on the 7th day, we miss about 6 days of click information.



33]:

	Impression_ID	User_ID	Time	History	Impressions	date	I
6511	6512	U56031	2019-11-11 12:45:40	N38105 N60702 N32799 N52555 N54536 N37091	N55204-1 N30998-0 N52622-0 N59981-0	2019-11-11	
10665	10666	U56031	2019-11-14 06:35:21	N38105 N60702 N32799 N52555 N54536 N37091	N47098-1 N50872-0 N46821-0 N23877-0 N38779-0 N...	2019-11-14	
26156	26157	U56031	2019-11-12 08:04:02	N38105 N60702 N32799 N52555 N54536 N37091	N52122-1 N50675-0 N41020-0 N49180-0 N7319-0 N3...	2019-11-12	
31994	31995	U56031	2019-11-13 06:10:01	N38105 N60702 N32799 N52555 N54536 N37091	N38159-0 N34876-1 N35233-0 N62128-0 N13907-0 N...	2019-11-13	

Figure 4: Behaviour data structure

Solution

In the development phase, we just need to predict next one day's data. So I use the traditional method to build a ctr prediction lightgbm model. But it didn't work in the test phase. So my submitted solution is mainly depend on the official model NAML , NRMS and NPA. I think these models are all good solution to solve the news recommendation question. And it is practicable. I will introduce the NAML and NRMS briefly.

NAML is a approach for neural news recommendation. There are three major modules, i.e., a news encoder with attentive multi-view learning to learn representations of news, a user encoder with attention mechanism to learn representations of users from their browsed news, and a click predictor to predict the probability of a user browsing a candidate news article. The architecture of NAML is shown in Fig.5.

2.

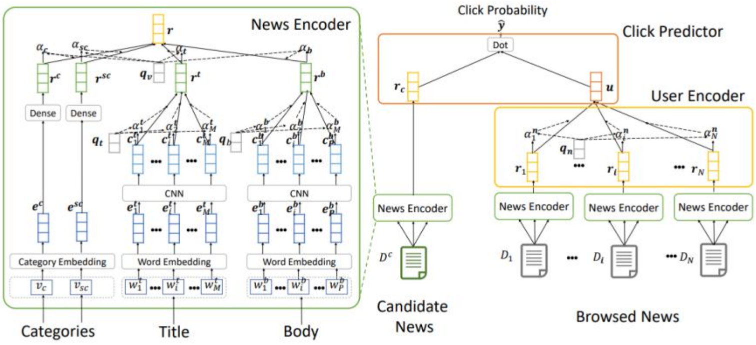


Figure 5: The framework of NAML

The core of NRMS's approach is a news encoder and a user encoder. The news encoder learn news representations from news titles by using multi-head self-attention to model the interactions between words. The user encoder learn representations of users from their browsing by using multi-head self-attention to capture their relatedness. Besides, it apply additive attentions to both news and user encoders to select important words and news to learn more informative news and user representations.

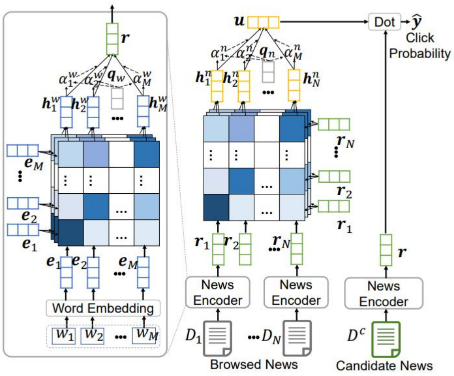


Figure 6: The framework of NRMS

For getting a good score on the leaderboard. I try the following 4 point modifications. It helps me improve the AUC score.

1. No valid set, Use all data to train the model.
2. Use first 50 history click behavior to train the model instead of last 50 history click behavior.
3. Use W2V pretrained representation to replace the GloVe embedding.
4. Local normalization method.

As mentioned above the data structure will lead to a big gap, and this gap will serious affect the Model train. So I try to use the first 50 click behaviors instead of the last 50 click behaviors. W2v embedding method is different from glove embedding method. it will lead to different result. So I choose this method to get diversity results. Table 1 shows the AUC score comparison on different method. Although we cannot get obvious improvement by modification. But I get promote by weighted sum method. Although the best score come from the Original Model. But I find the different results have some gap. By weighted sum methods on same Models. I got almost 0.69 AUC score as table 2 shown.

Improvement	NAML	NRMS
Original (6 epochs, train set and dev set)	0.6876	0.6862
First 50 history news replace last 50 news	0.6874	0.6858
W2V replace glove	0.6817	0.679

Table 1: Auc score on different method

NAML training History News	AUC
First 50	0.6876
Last 50	0.6874
Weighted sum the two above results.	0.6897

Table 2: Improvement from result fusion

But Different Model result have different distribution. We cannot directly use the weighting sum to blend two different Models results.

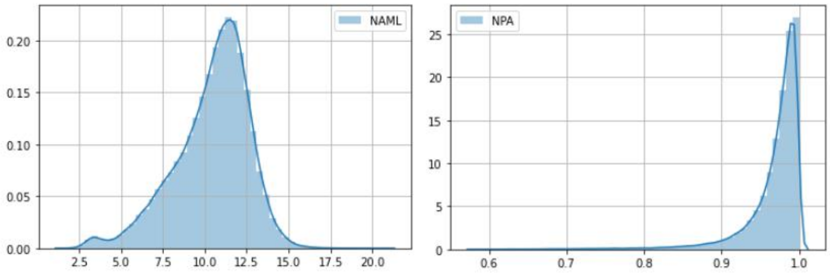
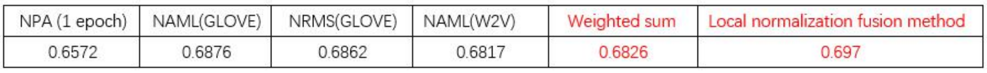


Figure 7: Different distribution from different models

Normalization is a simple method to solve this problem. We can use it to change the data’s distribution. But if we apply the global normalization method to the result, it will damage the results in single news impression list. To solve this question, here I thought a tricky Method to fuse two different results. For simple, I will call this method local normalization fusion method. The implementation as follows:

$$\sum_{i}^{n}\sum_{j}^{m} w_{i} * f(res_{ji}),\ f(x) = (x - avg(x)) / std(x)$$

Finally, by local normalization method it helps me fuse three different model from 0.6832 to 0.697.



NPA (1 epoch)	NAML(GLOVE)	NRMS(GLOVE)	NAML(W2V)	Weighted sum	Local normalization fusion method
0.6572	0.6876	0.6862	0.6817	0.6826	0.697

Table 3: Score by local normalization fusion method

Conclusion

In conclusion, I consider the official model is a good solution to solve the news recommendation question. We can get a nice score from it. And it is practicable to apply on the product. We can use it as a well framework and do some modification to transfer other task. In this competition, I put a new method to fuse different model results in recommendation field. It works very well to fuse different source's results.

Finally, I want to thanks the organizers again for hosting such a good competition. Thanks to my girlfriend - xiaoxiao jie. I cannot make it without her.