Multi-Objective Recommender System

OTTO 2022 — WSM Project 3

WSM_UTF8 | 2023/01/14
111753229 何子安 | 111753152 王良文 | 111753162 謝非諭 | 111753213 江昀紘

1. Abstract

1.1 A Brief Description Of The Final Project

This final project is related to a Kaggle competition named 'OTTO' which aims to develop a Multi-Objective Recommender System.
The link to the competition's dataset is attached below:
https://www.kaggle.com/competitions/otto-recommender-system/data.
In this competition, our team is utilizing various methods including Exploratory Data Analysis (Eda), ensemble methods, word2vec, matrix factorization and item-based collaborative filtering (itemcf).

1.2 Goal

The goal of this competition is to predict e-commerce clicks, cart additions, and orders. You'll build a multi-objective recommender system based on previous events in a user session.
Our work will help improve the shopping experience for everyone involved. Customers will receive more tailored recommendations while online retailers may increase their sales.

2. Introduction

2.1 Ensemble Method

Ensemble methods are techniques that aim at improving the accuracy of results in models by combining multiple models instead of using a single model. The combined models increase the accuracy of the results significantly. This has boosted the popularity of ensemble methods in machine learning.

2.2 Word2vec

Word2vec is one of the Word Embedding methods and belongs to the field of NLP. It is the process of converting words into "computable" and "structured" vectors. After the training is complete, the word2vec model can be used to map each word to a vector, and the vector can represent the words' relationship with other words.

2.3 ItemCF

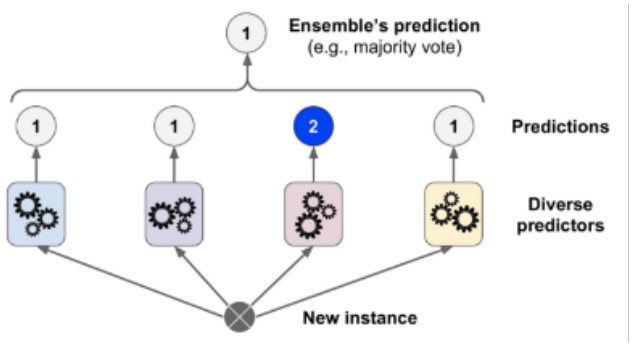
Item-based collaborative filtering (IBCF) was launched by Amazon.com in 1998, which dramatically improved the scalability of recommender systems to cater for millions of customers and millions of items. Prior to the launch of IBCF, there had been many systems of user-based collaborative filtering (UBCF) developed in the academia and the industry, which had the issues of huge computational cost and limited scalibilty, but since the IBCF algorithm was published in IEEE Internet Computing in 2003, it has been widely adopted across all the Web giants, including YouTube, Netflix, and lots of others. This article will bring a toy example to explain how UBCF and IBCF work and why internet giants prefer IBCF to UBCF.

2.4...
2.5...

3. Method

3.1 Ensemble Method

In this Kaggle:
https://www.kaggle.com/code/karakasatarik/0-578-ensemble-of-public-notebooks/notebook
The method like Voting!
The voting ensemble method is a type of ensemble method that combines the predictions of multiple models by Voting. The voting ensemble method can be used to make more accurate predictions than any single model by combining the knowledge and expertise of multiple experts. The idea is that, by pooling the predictions of multiple models, you can reduce the variance and avoid overfitting. The voting ensemble method is typically used when there are multiple models with different configurations, or when there are multiple experts with different opinions. In either case, the voting ensemble method can help to produce a more accurate prediction by aggregating the information from multiple sources. The picture below represents voting ensemble method:



3.2 Word2vec

In this Kaggle: https://www.kaggle.com/code/radek1/word2vec-how-to-training-and-submission
We can regard the product actions as sentence (like 25_clicks or 551_carts…),then converting the sentence into vectors, so that we can use the vectors to predict the goals. This notebook use “gensim” library to train the word2vec model because it offers extremely fast training on the CPU.Next it use “polars”to load and process the data.Before training the model we need to use “polars. groupby.agg()” to group the sessions with same aids,then use “polars.to_list()” to transform the data into array structure to feed the word2vec model.
I use gensim.word2vec() model with parameters (workers=8, window=9, vector_size=64, sg=0),workers means threads to train the model,window means maximum distance between the current and predicted word within a sentence,vector_size means dimensionality of the

word vectors,sg means training algorithm: 1 for skip–gram; otherwise CBOW.I've tried many sets of parameters and above has the highest score in kaggle with 0.519.
CBOW(Continuous Bag Of Words): The CBOW model tries to understand the context of the words and takes this as input. It then tries to predict words that are contextually accurate.
When outputting the submission i use "annoy" to find the neighbors with "Euclidean" to predict the goal. If the output fewer than 20, we can find the nearest neighbor to keep predict the output.

3.3 Item–Based Collaborative Filtering

Step 1: transpose the user–item matrix to the item–user matrix
As the item similarity is required by IBCF, the item–user matrix shown in the table, transposed from the corresponding user–item matrix, makes it more clear by viewing each row as an item vector during the similarity calculation.
Step 2: Calculate the similarity between any two items and fill up the item–item similarity matrix
Calculate the cosine similarity.

$$sim(i,j) = \cos(\vec{i},\vec{j}) = \frac{\vec{i}\cdot\vec{j}}{\|\vec{i}\|_2 * \|\vec{j}\|_2}$$

Formula 3: Cosine similarity used for calculating item similarity

Step 3: Predict the ratings
After successfully building the item–item similarity matrix, the calculation of can be done by injecting the values to Formula 4.

$$P_{u,i} = \frac{\sum_{\text{all similar items, N}}(s_{i,N} * R_{u,N})}{\sum_{\text{all similar items, N}}(|s_{i,N}|)}$$

Formula 4: Prediction formula for IBCF

I've tried many sets of parameters and above has the highest score in kaggle with 0.517.

3.4
3.5..
..
.

4. Result

4.1 About Kaggle's Score

Model	Try	Ensemble 1	Ensemble 2
Ensemble	0.575	0.542	0.554
	0.57		
	0.522		
	0.493		
Word2Vec	0.519		
ItemCF	0.517		
Marie Factorization	0.499		

5. Conclusion

5.1 Difficulties and Learned

1. Data preprocessing matter, which is har
2. Tomato better use of the TOP model, require model's fundamental as well
3. Ensemble and Learning to rank usually bring performance to next level
4. Limitations of hardware
    1. There is always bigger
    2. TFRecord
5. Stand on shoulder of the giant

5.2 About ItemCF

Offline calculation is the biggest advantage of item–based collaborative filtering in comparison with user–based collaborative filtering. However, the training process of the traditional item–based collaborative filtering is stagnant, which may take weeks for large datasets. Fortunately, there are some new researches, which remarkably reduces the computational cost, e.g. Sparse Linear Methods.

Reference

1. Page 497, Data Mining: Practical Machine Learning Tools and Techniques (Morgan Kaufmann Series in Data Management Systems)
2. https://analyticsindiamag.com/the–continuous–bag–of–words–cbow–model–in–nlp–hands–on–implementation–with–codes/
3. https://www.kaggle.com/code/s107304004/itemcf/notebook