Contrastive Learning for Event Sequences with Self-Supervision on multiple domains

Team 2:

Nikolay Kotoyants
Ivan Gurev
Gleb Mazanov
Anna Iliushina
Viacheslav Naumov

Motivation

Convert the specific data to a form that can be used for the classification task

The use of CoLES embeddings significantly improves the performance of existing models in downstream tasks and provides significant financial benefits. It is necessary to compare CoLES on several public datasets of event sequences and prove that CoLES representations consistently outperform other methods in various tasks.

Problem statement

- Assume there are some entities and that each entity's lifetime activity is observed as a sequence of events
- Each entity is a latent class, which is associated with a distribution over its possible samples and we observe only a single finite realisation
- Our goal is to learn an encoder that maps event sequences into a feature space in such a way that the obtained embedding encodes the essential properties of entity and disregards irrelevant noise contained in the sequence

Work plan

01 Resources

02 Preprocessing

03 Encoding

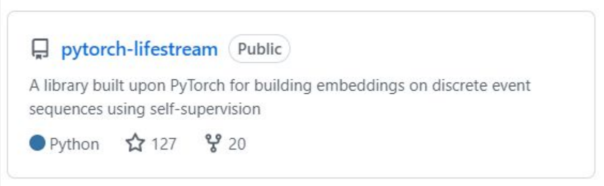
04 Classification

RESOURCES

1. Relevant paper

CoLES: Contrastive Learning for Event Sequences with Self-Supervision

2. GitHub library



3. Datasets from the competition

Data Fusion Contest 2022. The Education Challenge

Datasets

transaction.csv

	user_id	event_time	mcc_code	currency_rk	transaction_amt
0	000932580e404dafbecd5916d4640938	1.596442e+09	5411	48	-361.07230
1	000932580e404dafbecd5916d4640938	1.596591e+09	5499	48	-137.31398

clickstream.csv

	user_id	event_time	cat_id	new_uid
0	000a8d3cdef3455d990e97730a2cef43	1.611327e+09	12	1364191
1	000a8d3cdef3455d990e97730a2cef43	1.611429e+09	931	531108

Convert dataframe to a list of dictionaries, where the keys are the features

```
{'user_id': '000a8d3cdef3455d990e97730a2cef43',
 'new_uid': tensor([1364191,  531108,  531108,  ...,  617687, 1478288, 1364191]),
 'event_time': tensor([1.6113e+09, 1.6114e+09, 1.6115e+09,  ..., 1.6278e+09, 1.6279e+09,
         1.6281e+09], dtype=torch.float64),
 'cat_id': tensor([49,  3,  1,  ..., 20, 17, 19])}
```

train.csv

A target variable: the label of whether clients have higher education - 0 and 1

Encoders

1. CoLES
2. Random encoder
3. Agg baseline (AggFeatureSeqEncoder)

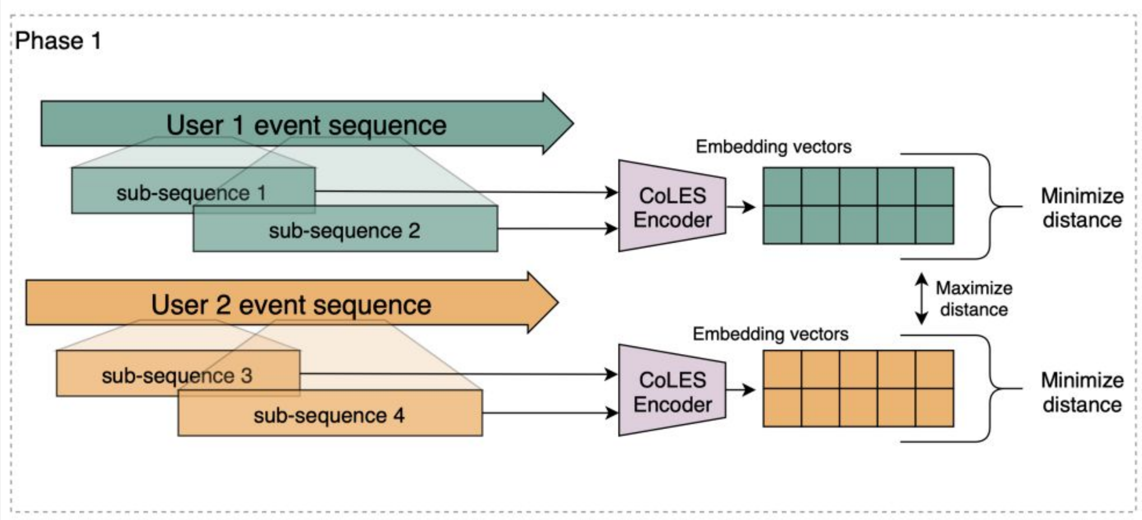
CoLES

The composite encoder model:

1. Event encoder: takes a set of attributes of each event and outputs its intermediate representation (linear layers, batch normalization layers)
2. Sequence encoder: takes the intermediate representations of the events and outputs the representation of their sequence up to the time

CoLES

Paper "CoLES: Contrastive Learning for Event Sequences with Self-Supervision"



Self-supervised training

Random encoder

1. The network architecture is the same as CoLEs
2. The weights are random
3. It seems that other networks should be just as good

Agg baseline

1. The categorical features are arranged in OHE and the numerical features are arranged in the resulting columns

2. The order of the transactions is not taken into account

3. A good baseline for the problem

Transaction dataset

We use RandomForest Classifier

Embedding	accuracy score	precision score	f1 score	Recall score	roc auc_score
CoLES	0.76	0.8	0.85	0.91	0.64
Random encoder	0.73	0.74	0.84	0.97	0.5
Agg baseline	0.78	0.79	0.86	0.95	0.62

Clickstream dataset

Embedding	accuracy score	precision score	f1 score	Recall score	roc auc_score
CoLES	0.72	0.72	0.84	0.99	0.5
Random encoder	0.69	0.72	0.81	0.92	0.49
Agg baseline	0.64	0.73	0.76	0.8	0.51

Conclusion

- Have implemented the methods presented in the article
- Got the result for 3 different encoders for 2 types of data
- Have proven that the result for CoLES is better than for other encoders

Thanks!

GitHub: https://github.com/fiestaxxl/ML-project