SNA 2016

team 005

Про конкурс

Социальный граф в виде UserID -> FriendID, Mask

Демография пользователей (дата рождения, дата создания аккаунта, пол, страна, регион)

1 млн. пользователей “ядра”

Есть “внешние” вершины без исходящих ребер

Для пользователей, у которых ID % 11 == 7, удалено 10% друзей

Про конкурс

Конкурс проходил в 2 этапа: online и offline

На втором этапе дают доп.данные: информацию о группах, демографию всех пользователей, взаимодействия.

Заслать нужно не более 4 млн. каднидатов.

Метрика: ndcg * 1000

Не было деления лидерборда на private/public

Оценивались презентация, продакшновость решения, идеи и ndcg

Готовим данные

Проблема:

Очень много данных (10^12 пар)

Нужно набрать кандидатов: юзеры на расстоянии 2 от данного

Трейн тест валидэйт (3 штуки) тини

Кандидаты для обучения/предсказания

1) Из нетестовых вершин удаляем 10% ребер
2) Запускаем функции, набирающие кандидатов по всему графу (80% ndcg)
3) Рассмотрим пару юзер-кандидат:
   Юзер является тестовым? Это объект, для которого мы должны предсказать
   Нет? Это объект для обучения

   Пара юзер-кандидат является удаленным ребром? Это “+” пример
   Нет? Это “-” пример
4) Берем 0.001 часть негативных примеров

Кандидаты в трейн и тест из одинаковых распределений
Значения фичей для трейна и теста тоже

Итого

40 млн пар - трейнинг сет (+ и - поравну примерно)

2 млрд пар - для предикта

Факторайзер

Мы хотим:

1) Делать очень много фичей
2) Продакшн решение (конвейер, воспроизводимость)
3) Использовать этот код и дальше
4) Считалось на мапредьюсе

Факторайзер

```
def __call__(self, user, candidates):

    return {"user['id'] candidates[0]['id']": [3, 44],

            "user['id'] candidates[1]['id']": [3, 55]}

def __call__(self, user):

    return {"user['id']": [3, 44],

             "user['id']": [3, 55]}
```

Факторайзер

Разные фишки:

- Передача ребер, демографии, svd векторов и других доп.данных на машинки (мемори маппинг)
- Построение графов на машинках (networkx)
- Убивание джобов по таймауту (забиваем фактор дефолтными значениями)
- Параллельный запуск транзакций

Счетчики

```
def __call__(self, user, candidates):

    return {"user['id'] candidates[0]['id']": ["M 0.25 F 0.45", "2012 RU 2013 RU"],

            "user['id'] candidates[1]['id']": ["F 0.65 F 0.65", "2010 RU 2014 FR"]}
```

Типы графов

1) Весь граф
2) Весь граф без “внешних вершин”
3) Граф по юзеру (окрестности 2)

   Самые естественный, выбираем N ребер на уровне 2

4) Граф по юзеру и кандидату (окрестности 1)
5) Граф по общим друзьям юзера и кандидата

Что обучаем?

Xgboost на классификацию

Давайте пилить факторы!

Факторы факторы факторы … ещё больше факторов

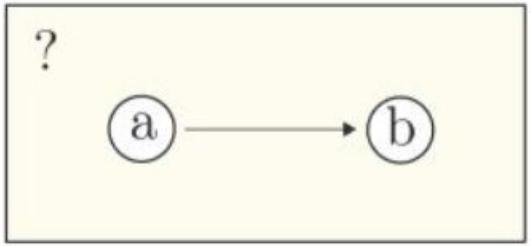
Количество всех факторов: ~240

После feature selection: ~85

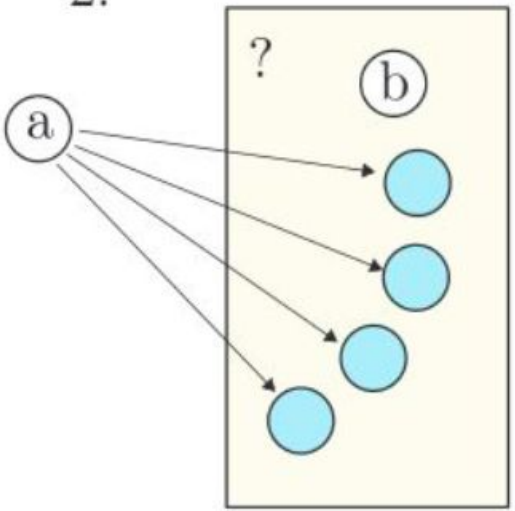
Сабмит №1: простые факторы

- Common friends
- Jaccard distance
- Cosine distance
- Preferential attachment
- Adar

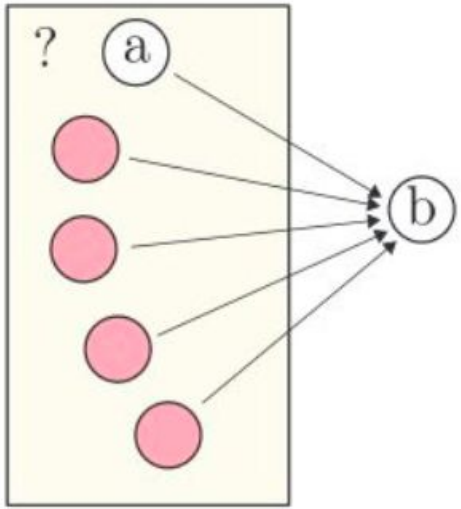
1.



2.



3.



Сабмит №2: аггрегации

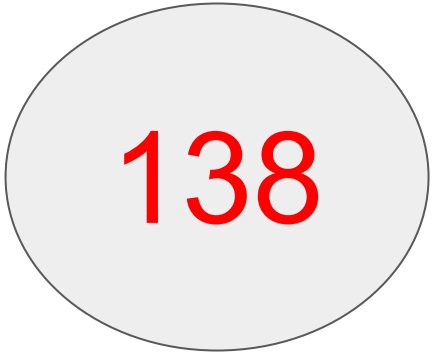
- …
- In aggregator: common friends
- In aggregator: jaccard distance
- In aggregator: cosine distance
- In aggregator: preferential attachment
- Out aggregator: common friends
- Out aggregator: jaccard distance
- Out aggregator: cosine distance
- Out aggregator: preferential attachment

Сабмит №3: демография

- …
- Age left/right
- Gender left/right
- External friends left/right
- Is equal login region
- Count of country id / location id / login region id for left and right
- Resource allocation index
- Create date
- Birthday

Сабмит №4: счётчики

- …
- Age pairs
- Gender pairs (Example: M 0.75 F 0.5)
- Friends mean gender
- Age + Gender pairs
- ...



Сабмит №5: EdgeRank + Triangles

- …
- Subgraph EdgeRank
- Subgraph EdgeRank reversed
- Subgraph Edgerank normed rank
- Subgraph triangles
- Subgraph triangles reversed

Сабмит №6: Subgraph clustering

- …
- Vertices count in candidate cluster
- Count of clusters
- Density
- Transitivity
- Distance between clusters (count of edges from user cluster to candidate cluster ...)

Сабмит №7: PageRank

- …
- PageRank left / right
- Subgraph pagerank left / right

Сабмит №8: SVD

- …
- SVD cosine (users/items)
- SVD pearson correlation (users/items)
- SVD euclidean distance (users/items)
- In aggregator SVD distances (users/items)
- Out aggregator SVD distances (users/items)
- Inner product (user left, item right)
- ...

Сабмит №9: Что засылать?

Ручная формула, выбирающая топ - 4000000 кандидатов.

- Берем не меньше 15 кандидатов на человека
- Но не больше 45
- А еще не больше 2 * количество удаленных
- А еще, если остались кандидаты с предиктом > 3.5, то тоже неплохо...
- ....
- PROFIT!
-

Спасибо за внимание!

Вопросы?