САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Математико-механический факультет

Кафедра информационно-аналитических систем

Курсовая работа по теме:

«Поиск связанных товаров в клиентской корзине (рекомендательная система)»

Студент: Лучко Александр Юрьевич

Научный руководитель:

к. ф.-м. н., доцент Графеева Н. Г.

Оглавление:

Введение

В век массового потребления практически все мы ходим в магазин или используем интернет технологии для покупок. Каждый владелец магазина хочет максимизировать свою выручку, для этого ему было бы неплохо продавать покупателю не только ту вещь, за которой он пришёл “нацелено”, но и дать “в нагрузку” ещё парочку товаров.
Но не только это, также продавец, хотел бы наблюдать зависимость покупаемых товаров, к примеру, знаменитое утверждение:

- If someone buys diaper and milk, then he/she is likely to buy beer

То есть если кто-то покупает подгузники и молоко, он, скорее всего, купит и пиво. Поэтому не плохо бы знать все такие зависимости и использовать их. Чем это может пригодиться?

- Ставить такие релевантные товары рядом, чтобы повысить ещё более вероятность покупки вместе.

- Давать скидку на одни товары, но повышать на другие
  купил: {подгузники, молоко}→ купит: {пиво}
  На подгузники скидка, но пиво очень дорогое.

- Подобный анализ может дать более наглядную картину на бизнес в целом.

Постановка задачи

Целью данной работы является реализация приложения, которое строит анализ клиентской корзины, может рекомендовать пользователю купить что-то ещё помимо текущих его покупок (рекомендательная система).

Так как данная работа преследовала цель получить готовый продукт, который выполняет данные функции, описанные выше, была проведена декомпозиция задач.

Список действий для реализации системы:

• Чтение пользовательского набора csv файлов.

• Перевод csv файлов в объекты базы данных.

• Перевод данных в более компактный вид для хранения и работы.

• Построение рекомендательной системы на основе собранных данных .

Краткое описание каждого шага декомпозиции

Чтение пользовательского набора csv файлов
файл имеет такую структуру

CliCode	RgdCode	RgdGroup	TerrCode	RgdQuant	QuantCapt
1	466014	16	6316	2	1
1	514923	13	6316	1	1
1	704407	14	6316	3	1
1	828723	11	3400	70	1

Перевод csv файлов в объекты базы данных

Была использована SQLite реляционная база данных. Структура таблиц полностью повторяет структуру csv файлов.

Перевод данных в более компактный вид для хранения и работы

Хранение происходит в таком виде (смотрите пример ниже), а именно есть вектора для каждого пользователя в пространстве размерности количества всех товаров. Если указанный пользователь брал товар, то в его векторе будет стоять 1 на месте (координате) данного товара, иначе 0.
Пример, пространство размерности 5 товаров.
Клиент 1 имеет такой вектор

```
[1       ,0       ,0        ,1       ,0       ]
[product1,product2, product3,product4,product5]
```

значит, клиент 1 взял **product1, product4.**
Дальше данная структура часто будет называться продуктовой корзиной пользователя (клиента), а также вектором в пространстве всех товаров, оба определения эквивалентны.

Замечание:

Так как товаров очень много, а один пользователь может взять всего лишь один товар, то строить вектора в векторном пространстве размерности количества всех товаров слишком дорого по памяти, поэтому были использованы csr_matrix (разряженные матрицы).

Построение рекомендательной системы на основе собранных данных

Алгоритмы, которые будут рассматриваться:

A-Priori Algorithm (программно не реализован)

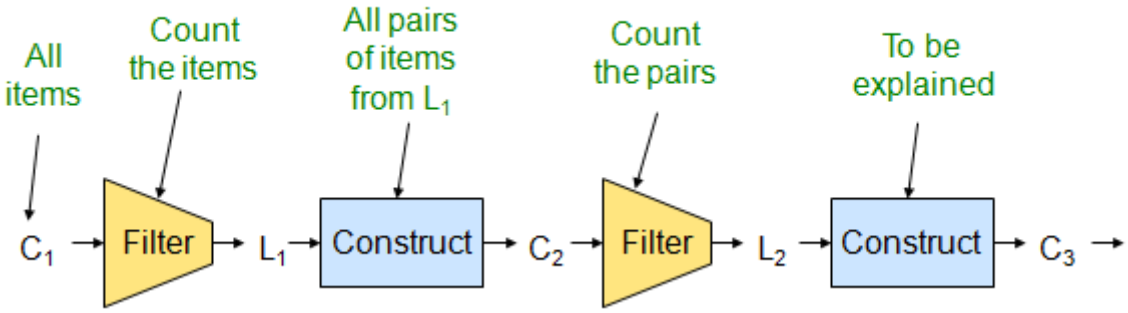
Выделение правил на графе связанности товаров (программно реализован).

Ниже будет обговорено более конкретно для каждого алгоритма.

A-priori алгоритм

Кратко, мы ищем часто встречающие наборы товаров, смотря на предыдущие шаги

- **For each *k*, we construct two sets of *k*-tuples** (sets of size *k*):
  - C_k = *candidate* ***k*-tuples** = those that might be frequent sets (support ≥ **s**) based on information from the pass for ***k*–1**
  - L_k = the set of truly frequent ***k*-tuples**



- **Hypothetical steps of the A-Priori algorithm**
  - C_1 = { {b} {c} {j} {m} {n} {p} }
  - Count the support of itemsets in C_1
  - Prune non-frequent: L_1 = { b, c, j, m }
  - Generate C_2 = { {b,c} {b,j} {b,m} {c,j} {c,m} {j,m} }
  - Count the support of itemsets in C_2
  - Prune non-frequent: L_2 = { {b,m} {b,c} {c,m} {c,j} }
  - Generate C_3 = { {b,c,m} {b,c,j} {b,m,j} {c,m,j} } **
  - Count the support of itemsets in C_3
  - Prune non-frequent: L_3 = { {b,c,m} }

После этого можем провести декомпозицию корзины и пробовать найти правила покупки.

$B_1 = \{m, c, b\}$ $B_2 = \{m, p, j\}$

$B_3 = \{m, b\}$ $B_4 = \{c, j\}$

$B_5 = \{m, p, b\}$ $B_6 = \{m, c, b, j\}$

$B_7 = \{c, b, j\}$ $B_8 = \{b, c\}$

- **Association rule: {m, b} → c**
  - **Confidence** = 2/4 = 0.5
  - **Interest** = |0.5 – 5/8| = 1/8
    - Item ***c*** appears in 5/8 of the baskets
    - Rule is not very interesting!

Плюсы:

1)нахождение правил.
2)простота понимания алгоритма в целом
3)настройка алгоритма под dataset

Минусы:

1)сложности работы с памятью.
2)индивидуальный подход к каждому набору данных

Выделение правил на графе связанности товаров

Я решил воспользоваться подходом A-priori алгоритма и поискать правила и подобную информацию. Источником анализа я выбрал графы, построенные по алгоритму ниже.

1. Вершины - товары (неориентированный граф)
2. Пробегаем по всем клиентским корзинам, если два товара лежат в одной корзине, добавляем ребру между ними +1(делаем это для полученной пары товаров всего раз на одной корзине). Граф, построенный на шаге 2, назовём *“первичным графом связанности товаров”*.
3. Подсчитываем количества взятия каждого товара по всем клиентам (можно совместить с шагом 2). Данный массив так и назовём “ массив взятия товаров”.
4. Строим ориентированный граф связанности товаров из графа, полученного на шаге 2. Просто, удваивая ребро между вершинами и ставя в оба направления вес изначального ребра.
5. Делим каждое ребро, исходящее из данной вершины (товара), на количество взятий этого товара.
6. Заметим, что вес каждого ребра станет число из (0, 1). Удалим все рёбра, чей вес меньше заранее заданного числа CHRG. Граф, полученный на шаге 6, назовём *“конечным графом связанности товаров”.*
7. Точно также как на шаге 6, получим *“первичный нормированный граф связанности товаров”.* Сделав с первичным графом товаров (который неориентированный) операцию нормирования, а именно поставим такой вес на рёбро,
   протянутое между вершинами A и B:
   2*(текущий вес)/(количество взятия A + количество взятия B). Очевидно, что это число лежит в промежутке (0, 1). Удалим все рёбра, чей вес заранее меньше заданного числа CHRG, который тоже лежит в (0, 1).

Замечание:

откидывание нужно, чтобы удалять не интересующие нас рёбра, которые в свою очередь характеризуют покупку вместе данных товаров. Мы оставляем только те события, которые имеют по нашей нормировки или в стандартном смысле высокую вероятность на происхождение (нижней предел этой вероятности мы выбираем сами). Тем самым найденные правила, которые, по сути, являются внутренними структурами на графах (такими как циклы, клики и тд) также будут иметь высокую вероятность на происхождения, то есть благодаря откидыванию, мы получаем лишь достаточно вероятные события на нашем dataset.

Описание полученных графов:

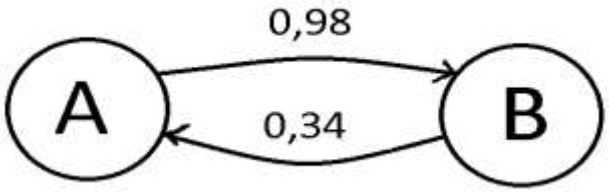
“первичный нормированный граф связанности товаров” - граф, характеризующий вероятностную связь взятие товаров, то есть, если товары почти всегда берутся вместе, то значением CHRG относительно близким к 1 они пройдут фильтрацию (их не удалят). Можно выбирать иные методы нормировки, вкладывая в это характеристику те критерии, которыми мы заинтересованы, к

примеру, деление на 2max(взятие A,взятие B) и тд. В общем, существует множество других нормировок на данном графе. В данной нормировке можно увидеть недостаток, который полностью покрывается конечным графом связанности товаров, а именно, если один товар (назовём его A), берётся с другим (назовём его B) почти всегда, однако, B может браться ещё с кучей других товаров, покупки с товаром A составляют у него только малую часть, данная нормировка может не пропустить ребро между ними, то есть рёбро между A и B на шаге 7 может быть откинуто, хотя связь между товарами A и B есть, и мы хотим рекомендовать B, когда покупают A.
Оставленное же ребро между товарами (в данном построении) будет говорить о том, что связь сильная в обе стороны.

“конечный граф связанности товаров” – граф, характеризующий направления отдачи данной вершины (товара) своего потенциала покупки с другим товаром. Поясню, к примеру, если есть вершины A и B и между ними ребро, исходящие из A, вес этого ребра 0.3, это означает, что товар A, будет браться с товаром B в 30% случаях (30% процентов всех покупок A, будут вместе с покупками товара B). Такой граф будет показывать даже такие зависимости, к примеру, есть мобильный телефон специфическим для него зарядным устройством, которое продаётся в официальных магазинах данного бренда, берут же его почти всегда при первоначальной покупке телефона, как отдельный товар. Во всех остальных случаях люди заказывают зарядные устройства других производителей ввиду экономии средств. Получаем, что на ребре от зарядного устройства к телефону будет стоять число близкое к 1, но на обратном ребре, в принципе может стоять что угодно (есть те, кто не берёт зарядное устройство при покупке телефона), такое могло не пройти будь граф не ориентированный, в данном же случае, мы можем потерять при откидывании только ребро от телефона к зарядному устройству, но высоковероятное событие в обратную сторону не будет потеряно.

Характеристика связей на конечном графе связанности товаров

Обозначим телефон за B, зарядное устройство за A.

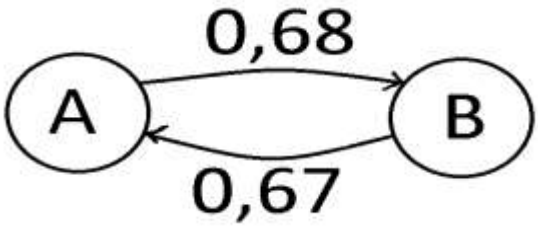


Пусть мы получили такие вероятности. То есть A, берут с B с вероятностью 0.98, а B с A с вероятностью 0.34. Было бы глупостью рекомендовать товар B при покупке товара A, так как их стоимости не соизмеримы, но рекомендовать товар A при покупке B имеет смысл. Поэтому, при поиске таких товаров, которые как A практически всегда берутся с B, и B имеет несоизмеримо большую стоимость, надо давать рекомендацию именно покупателем B, найти её сможем по входящему ребру в B, которое пройдёт просеивание (с CHRG<0.98).

Замечание:

Можно было бы понимать рёбра в инвертируемом смысле, подразумевая под этим более удобную картину восприятия “товар не отдаёт свой потенциал, товар принимает свой потенциал”, в любом случае конечный анализ не изменится, кому как нравится.

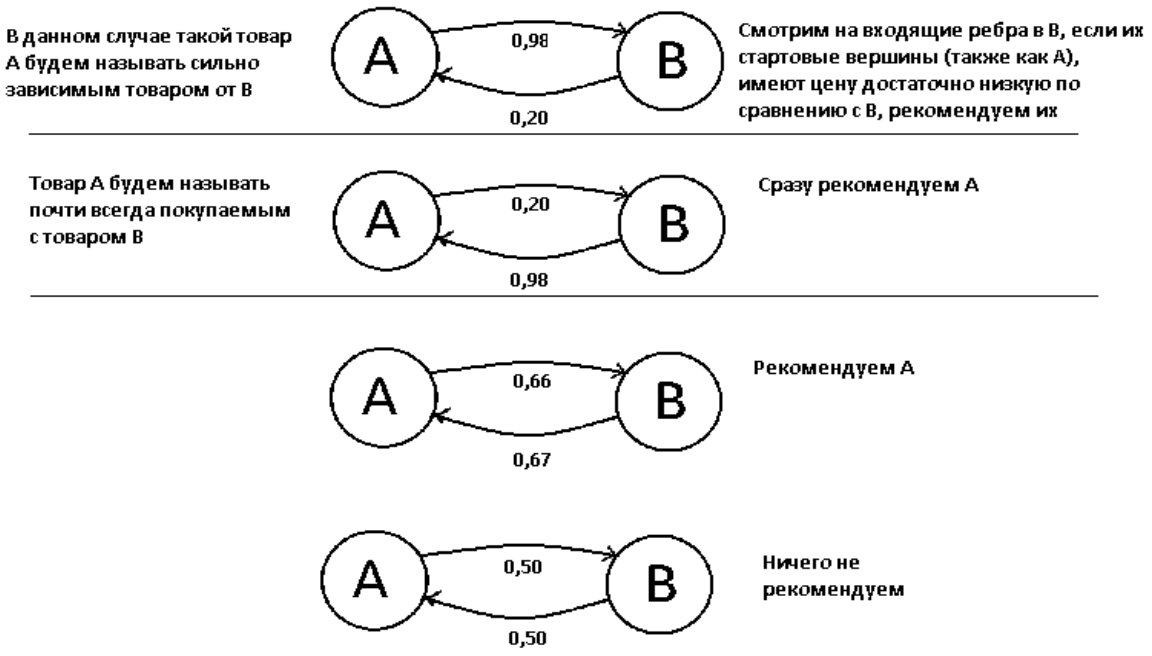
А теперь посмотрим на такую связь



Если при этом А также зарядное устройство, а В телефон. Оба пройдут просеивание с CHRG<0.67, но точно также нельзя будет рекомендовать при покупке А покупать В (очевидно!).
Так как же относиться к таким связям?
Будь это одно ребро или цикл длины два (или длины 3 и тд) найти максимальную стоимость в этой группе товаров и развертывать путь по вершинам, начиная с неё. В данном случае рекомендовать А при покупке В.

Краткое описание поведения в таких случаях на примере

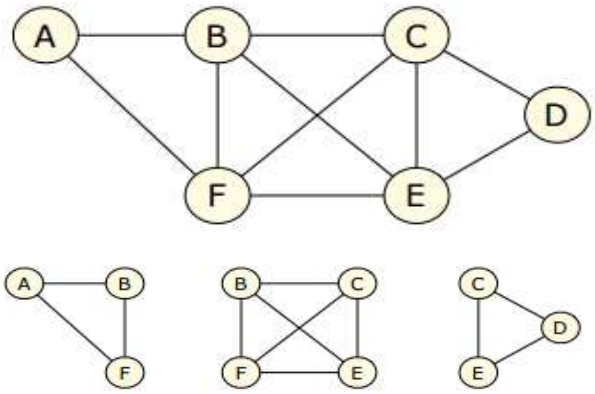
Пусть CHRG=0.65, цена А=120руб. Цена В=3000руб.



при относительно равных ценах, А может также рекомендовать и В. Для упрощения этой модели можно построить ценно зависимые веса рёбер, производить также отсеивания и уже не рассматривать случаи, приведенные на картинке выше.

Что можно делать с первичным нормированным графом связанности товаров.

1)Искать клики.



На картинке представлен связный граф, но связность не гарантируется – это просто пример

Теперь имея эти правила “правила на покупки”, мы можем рекомендовать пользователям, которые купили A,B, купить ещё товар F. Данные правила характеризуются очень сильной зависимостью товаров по клике в целом.

2)кластеризация на графах

Тем самым можно также выделить группы связанных товаров.

Что можно делать с конечным нормированным графом связанности товаров.

1)искать циклы

Это и будут правила вида.

- If someone buys diaper and milk, then he/she is likely to buy beer

ниже пример полученный на dataset

2)искать клики,

Более мощные связи - более жёсткие правила на покупку товара. Крепкое правило на клику в целом (результаты будут очень похожи с поиском клик на первичном нормированном графе связанности товаров).

Вывод

Всё это представляет огромный интерес и даёт обширное знание о зависимостях товаров. К примеру, благодаря этому методу, можно вытащить из данных информацию о клиентских корзинах (не ту что была показана выше), а именно закупать товар в нужных пропорциях, то есть, если фирма решит повысить продажи одного товара, то она может спрогнозировать эффективное повышение количества связанных товаров, тем самым максимизировать свою прибыль. Очевидно, всё это может работать и на прогнозирование нехватки товара и

переполнение товара на складах (надо дать вершинам вес, равный проценту остатка, анализ будет подобен предыдущим идеям).

Что было найдено на данном наборе данных

Представлена не вся вырезка! Данные товары покупаются с вероятностью 0,5 вместе.
метод: нахождение циклов на конечном графе связанности товаров.

```
[['Активное оборудование'] ['Клапаны и приводы'] ]
[['Светорегуляторы'] ['Панели оператора'] ]
[['Вертикальные стойки и опоры'] ['Комплекты "Крыша и основание"'] ['Задние панели'] ]
[['Вертикальные стойки и опоры'] ['Комплекты "Крыша и основание"'] ]
[['Вертикальные стойки и опоры'] ['Задние панели'] ['Комплекты "Крыша и основание"'] ]
[['Вертикальные стойки и опоры'] ['Задние панели'] ]
[['Задние панели'] ['Комплекты "Крыша и основание"'] ]
[['Вертикальные стойки и опоры'] ['Комплекты "Дверь и задняя стенка"'] ['Комплекты "Крыша и основание"'] ]
[['Вертикальные стойки и опоры'] ['Комплекты "Дверь и задняя стенка"'] ]
[['Вертикальные стойки и опоры'] ['Комплекты "Крыша и основание"'] ['Комплекты "Дверь и задняя стенка"'] ]
[['Вертикальные стойки и опоры'] ['Комплекты "Крыша и основание"'] ]
[['Комплекты "Крыша и основание"'] ['Комплекты "Дверь и задняя стенка"'] ]
[['Прочее оборудование'] ['Ответвители окрашенные RAL'] ]
[['Кронштейны к опорам для уличного освещения'] ['Граненые опоры для уличного освещения'] ['Закладные детали к ст
[['Кронштейны к опорам для уличного освещения'] ['Граненые опоры для уличного освещения'] ]
[['Закладные детали к стандартным опорам для уличного освещения'] ['Граненые опоры для уличного освещения'] ]
[['Светильники специального назначения'] ['Щиты серии ЩО-70'] ]
[['Выключатели нагрузки'] ['Вспомогательные элементы и аксессуары'] ]
[['Силовые разъемы'] ['Вспомогательные элементы и аксессуары'] ]
[['Электроинструмент'] ['Пилы, лобзики'] ]
[['Центробежные вентиляторы'] ['Дополнительное оборудование СКУД'] ['Светодиодные модули (LED)'] ]
[['Центробежные вентиляторы'] ['Дополнительное оборудование СКУД'] ]
[['Центробежные вентиляторы'] ['Светодиодные модули (LED)'] ['Дополнительное оборудование СКУД'] ]
[['Центробежные вентиляторы'] ['Светодиодные модули (LED)'] ]
[['Дополнительное оборудование СКУД'] ['Светодиодные модули (LED)'] ]
[['Коробки для пожароопасных помещений'] ['Перфодетали (уголки, пластины, опоры), хомуты'] ]
```

рассмотрим, к примеру, последнее правило. Коробки для пожарных помещений берут с вероятностью 0,5 в обе стороны с уголками, пластинами и опорами. Что абсолютно оправдано в реальной жизни.

2)Клики на неориентированном графе с CHRG=0.8. Представлена не вся вырезка.

```
[['Лампы автомобильные'] ['Электромагнитные ПРА'] ]
[['Лампы автомобильные'] ['Электронные ПРА'] ]
[['Лампы автомобильные'] ['Автоматические выключатели в модульном исполнении'] ]
[['Лампы автомобильные'] ['Лампы для светосигнальной арматуры'] ]
[['Боковые панели (стенки)'] ['Модули индикации и измерения'] ]
[['Элементы комплектации шкафов'] ['Консоли'] ]
[['Цоколи для шкафов и аксессуары к ним'] ['Модули индикации и измерения'] ]
[['Лампы для светофоров'] ['Арматура для монтажа СИП'] ]
[['Лампы для светофоров'] ['Кабели контрольные'] ]
[['Лампы для светофоров'] ['Провода для воздушных линий электропередачи (ЛЭП)'] ]
[['Аудиодомофоны'] ['Щиты распределительные металлические'] ]
[['Аудиодомофоны'] ['Кабели силовые для стационарной прокладки'] ]
[['Аудиодомофоны'] ['Щиты учетно-распределительные металлические'] ]
[['Аудиодомофоны'] ['Датчики'] ]
```

Список использованной литературы и источников:

1. http://www.mmds.org/
   CS246: Mining Massive Datasets chapter 6
2. http://mlg.ucd.ie/
3. http://scikit-learn.org/stable/modules/clustering.html
4. http://www.machinelearning.ru/wiki/images/2/28/Voron-ML-Clustering-slides.pdf
5. http://theory.stanford.edu/~sergei/slides/BATS-Means.pdf
6. http://infolab.stanford.edu/~ullman/mmds/ch7.pdf