

Автоматическое дополнение плейлистов в рекомендательной системе*

Кислинский В. Г., Фролов Е., Воронцов К. В.

kislinskiy.vg@phystech.edu; evgeny.frolov@skolkovotech.ru; vokov@forecsys.ru

Московский физико-технический институт

Работа посвящена исследованию метода совместной матричной факторизации в задаче top- N рекомендаций для автоматического продолжения плейлистов. Предлагается использовать модель матричной факторизации, учитывающая дополнительную информацию о плейлистах и треках. В данном методе латентное пространство приобретает дополнительную структуру, соответствующую признаковым описаниям плейлистов и треков. Также будет введена дополнительная регуляризация, основанная на предположении, что если объекты близки в пространстве признаков, то они также близки и в латентном факторном пространстве. Для анализа качества представленного алгоритма проводятся эксперименты на выборке из миллиона плейлистов MPD.

Ключевые слова: *задача top- N рекомендаций, совместная матричная факторизация, алгоритм LSE, латентное факторное пространство, коллаборативная фильтрация.*

1 Введение

Основной областью исследования в задаче автоматического продолжения плейлистов являются рекомендательные системы [1]. Ставится задача top- N рекомендаций, где пользователи — это плейлисты, объекты — треки. Рекомендательные системы имеют две основные группы методов: коллаборативная фильтрация и контентно-основанные (англ. content-based) методы. Контентно-основанные системы используют свойства объектов, например, автора, альбом, популярность треков. Создают профили пользователей, опираясь на свойства объектов, которые пользователь предпочел ранее, и рекомендуют новым пользователям наиболее подходящие к их профилю объекты. Коллаборативная фильтрация использует историю действий пользователей в системе для получения новых рекомендаций. Методы коллаборативной фильтрации основаны на предположении о том, что те, кто одинаково оценивали какие-либо объекты в прошлом, склонны давать похожие оценки в будущем. Такие методы устойчивы и способны выявлять скрытые свойства объектов, что улучшает релевантность рекомендаций. Матричная факторизация является одним из методов коллаборативной фильтрации. Каждому пользователю и объекту, сопоставляются некоторые факторы, которые определяют их скрытые интересы и свойства. Основным недостатком этого подхода является проблема холодного старта: неспособность строить рекомендации для новых пользователей и рекомендовать новые объекты.

В данной работе предполагается, что использование свойств объектов и пользователей улучшит качество рекомендаций, получаемых методами коллаборативной фильтрации и решит проблему холодного старта. Исследуется модель совместной матричной факторизации, подобный подход предлагается в работе [3], где матрица документ-термин факторизуются вместе с матрицей документ-пользователь. Существуют различные техники учета дополнительной информации в модели матричной факторизации, введение допол-

нительной регуляризации [4], совместная факторизация с матрицами похожести объектов и пользователей [5].

Основной целью работы является решение проблемы холодного старта для плейлистов. Методы матричной факторизации показывают хорошее качество рекомендаций для плейлистов, которые содержат достаточно большой набор треков, но для новых плейлистов, которые содержат мало треков или не содержат совсем, методы матричной факторизации строят рекомендации, которые имеют низкую релевантность. Если же строить рекомендации на основе свойств плейлистов и треков, то для новых плейлистов можно получить более подходящие рекомендации, например, был создан плейлист с именем «рок», тогда будет логичнее предложить наиболее популярные треки из рок-музыки, а не из другого жанра, но такой контентно-основанный подход строит слишком тривиальные рекомендации для плейлистов, которые уже имеют много треков. Совместная матричная факторизация объединяет в себе оба этих подхода, ищутся такие профили плейлистов, которые зависят не только от матрицы плейлист-трек, но и от матрицы плейлист-признак. Также рассматривается предположение о том, что из близости плейлистов в пространстве признаков, следует близость в латентном факторном пространстве, пространстве профилей, для проверки этого предположения вводится дополнительная регуляризация [3].

2 Постановка задачи

Задано множество треков $\mathcal{T} = \{t_i\}_{i=1}^m$ и множество плейлистов $\mathcal{P} = \{p_i\}_{i=1}^n$. Каждый трек описывается исполнителем и альбомом, всего k различных авторов, l различных альбомов, каждый плейлист имеет название. Определим матрицу $\mathbf{R} \in \mathbb{R}^{n \times m}$, следующим образом $\mathbf{R}_{ij} = 1$, если плейлист p_i содержит трек t_j , иначе ноль, также зададим бинарную матрицу $\mathbf{X} \in \mathbb{R}^{n \times (k+l+d)}$, которая описывает треки, каких авторов содержатся в плейлисте — k первых столбцов, каких альбомов — l следующих столбцов. Последние d столбцов — это векторное представление названий плейлистов. Надо для плейлиста p найти N наиболее подходящих треков, для этого будет строиться вектор $\mathbf{r} \in \mathbb{R}^{m \times 1}$, i -ый элемент которого означает насколько трек t_i подходит плейлисту p .

3 Описание метода

Задача матричной факторизации заключается в нахождении двух матриц меньшей размерности, произведение которых приближает исходную. В задаче рекомендаций ищется приближение матрицы рейтингов. В наших терминах

$$\mathbf{R} \approx \mathbf{U}\mathbf{V},$$

где \mathbf{U} — матрица профилей плейлистов, \mathbf{V} — матрица профилей треков. Предполагая, что профили плейлистов зависят от того, какие исполнители, альбомы входят в плейлист, какие названия у плейлистов, запишем

$$\mathbf{X} \approx \mathbf{U}\mathbf{H},$$

где \mathbf{H} — матрица, задающая малоразмерное латентное представление авторов, альбомов, названий. Приходим к задаче оптимизации:

$$\begin{aligned} \hat{\mathbf{U}}, \hat{\mathbf{V}}, \hat{\mathbf{H}} = \arg \min_{\mathbf{U}, \mathbf{V}, \mathbf{H}} & \alpha \|\mathbf{R} - \mathbf{U}\mathbf{V}\|_F^2 + (1 - \alpha) \|\mathbf{X} - \mathbf{U}\mathbf{H}\|_F^2 + \\ & \lambda (\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + \|\mathbf{H}\|_F^2), \\ \text{s.t. } & \mathbf{U} \geq 0, \mathbf{V} \geq 0, \mathbf{H} \geq 0 \end{aligned} \quad (1)$$

где λ положительный коэффициент регуляризации, а $\alpha \in [0, 1]$, если $\alpha > 0.5$, то матрица \mathbf{U} больше зависит от распределения треков по плейлистам, если $\alpha < 0.5$, то наоборот, предполагается, что \mathbf{U} зависит от распределения авторов, альбомов по плейлистам и названий плейлистов.

3.1 Введение дополнительной регуляризации на основе близости плейлистов в пространстве признаков

Близость плейлистов в пространстве признаков оценивается евклидовым расстоянием. Составим граф близости плейлистов, в котором вершинами будут плейлисты. Каждый плейлист соединим ребром с r ближайшими плейлистами, где r ранг разложения матриц \mathbf{R} и \mathbf{X} . По полученному графу составим матрицу смежности \mathbf{A} , где $\mathbf{A}_{ij} = 1$, если вершина i , соединена с вершиной j , ноль иначе. Предполагая, что если плейлисты близки в реальном пространстве признаков, то они также близки в пространстве профилей, с помощью матрицы \mathbf{A} определим близость, соответственных профилей:

$$\begin{aligned} S &= \frac{1}{2} \sum_{i,j=1}^n \|\mathbf{u}_i - \mathbf{u}_j\|^2 \mathbf{A}_{ij} = \sum_{i=1}^n (\mathbf{u}_i^T \mathbf{u}_i) \mathbf{D}_{ii} - \sum_{i,j=1}^n (\mathbf{u}_i^T \mathbf{u}_j) \mathbf{A}_{ij} = \\ &= \text{Tr}(\mathbf{U}^T \mathbf{D} \mathbf{U}) - \text{Tr}(\mathbf{U}^T \mathbf{A} \mathbf{U}) = \text{Tr}(\mathbf{U}^T \mathbf{L} \mathbf{U}), \end{aligned} \quad (2)$$

где \mathbf{D} диагональная матрица, на ii месте которой стоит сумма i строки матрицы \mathbf{A} , матрица $\mathbf{L} = \mathbf{D} - \mathbf{A}$. Перепишем (1), учитывая S :

$$\begin{aligned} \hat{\mathbf{U}}, \hat{\mathbf{V}}, \hat{\mathbf{H}} &= \arg \min_{\mathbf{U}, \mathbf{V}, \mathbf{H}} \alpha \|\mathbf{R} - \mathbf{U}\mathbf{V}\|_F^2 + (1 - \alpha) \|\mathbf{X} - \mathbf{U}\mathbf{H}\|_F^2 + \\ &\quad \beta \text{Tr}(\mathbf{U}^T \mathbf{L} \mathbf{U}) + \lambda (\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + \|\mathbf{H}\|_F^2), \\ \text{s.t. } &\mathbf{U} \geq 0, \mathbf{V} \geq 0, \mathbf{H} \geq 0. \end{aligned} \quad (3)$$

3.2 Задача оптимизации

Полученная задача оптимизации не является выпуклой по параметрам $(\mathbf{U}, \mathbf{V}, \mathbf{H})$. Следовательно поиск глобального минимума — затруднительная задача. Предлагается итеративный алгоритм поиска стационарной точки оптимизируемого функционала(3), который получен и доказан в статье [3]. Матрицы $(\mathbf{U}, \mathbf{V}, \mathbf{H})$ обновляются по следующим правилам:

$$\begin{aligned} \mathbf{U} &= \mathbf{U} \odot \frac{\alpha \mathbf{R} \mathbf{V}^T + (1 - \alpha) \mathbf{X} \mathbf{H}^T + \beta \mathbf{A} \mathbf{U}}{\alpha \mathbf{U} \mathbf{V} \mathbf{V}^T + (1 - \alpha) \mathbf{U} \mathbf{H} \mathbf{H}^T + \beta \mathbf{D} \mathbf{U} + \lambda \mathbf{U}} \\ \mathbf{V} &= \mathbf{V} \odot \frac{\alpha \mathbf{U}^T \mathbf{R}}{\alpha \mathbf{U}^T \mathbf{U} \mathbf{V} + \lambda \mathbf{V}} \\ \mathbf{H} &= \mathbf{H} \odot \frac{(1 - \alpha) \mathbf{U}^T \mathbf{X}}{(1 - \alpha) \mathbf{U}^T \mathbf{U} \mathbf{H} + \lambda \mathbf{H}}, \end{aligned}$$

где \odot - поэлементное умножение, \div - поэлементное деление.

3.3 Получение рекомендаций

Для того чтобы получить рекомендации для нового плейлиста, составим строку признаковов описания $\mathbf{x} \in \mathbb{R}^{1 \times (k+l+d)}$. С помощью метода наименьших квадратов из системы $\mathbf{x} = \mathbf{u} \mathbf{H}$ найдем профиль плейлиста \mathbf{u} . Умножая вектор \mathbf{u} на матрицу \mathbf{V} , получим вектор \mathbf{r} , i -ый элемент, которого означает насколько трек t_i подходит новому плейлисту, после этого из вектора \mathbf{r} выбирается top- N значений с индексами $\{i_1, \dots, i_N\}$ и рекомендуются треки $\{t_{i_1}, \dots, t_{i_N}\}$.

4 Оценка качества алгоритма

Качество полученных рекомендаций оценивается с помощью двух критериев: R-precision и $nDCG$ - normalized discounted cumulative gain. Первый из них показывает насколько, рекомендованные треки соответствуют интересам пользователя, второй помимо релевантности треков, показывает качество ранжирования треков, $nDCG$ увеличивается, если релевантно рекомендованные треки находятся выше в списке top- N рекомендаций. Пусть R список рекомендованных треков, G - список истинных треков.

R-precision определяется, как отношение правильно рекомендованных треков на количество истинных треков.

$$\text{R-precision} = \frac{|G \cap R_{1:|G|}|}{|G|}$$

Теперь определим $nDCG$,

$$nDCG = \frac{DCG}{IDCG},$$

где

$$DCG = 1 + \sum_{i=2}^{|R|} \frac{rel_i}{\log_2 i},$$

$$IDCG = 1 + \sum_{i=2}^{|R|} \frac{1}{\log_2 i}.$$

5 Базовый вычислительный эксперимент

Оценка качества алгоритма и сравнение с другими методами проводится на случайной подвыборке из миллиона плейлистов, содержащей 7657 плейлистов и 8560 треков, при этом каждый плейлист содержит не менее пяти треков. По данной подвыборке проводится следующая кроссвалидация: множество плейлистов разбивается на пять примерно равных частей, качество оценивается на каждой из пяти частей поочередно, на остальных обучается, при этом на тестовых плейлистах скрывается часть треков.

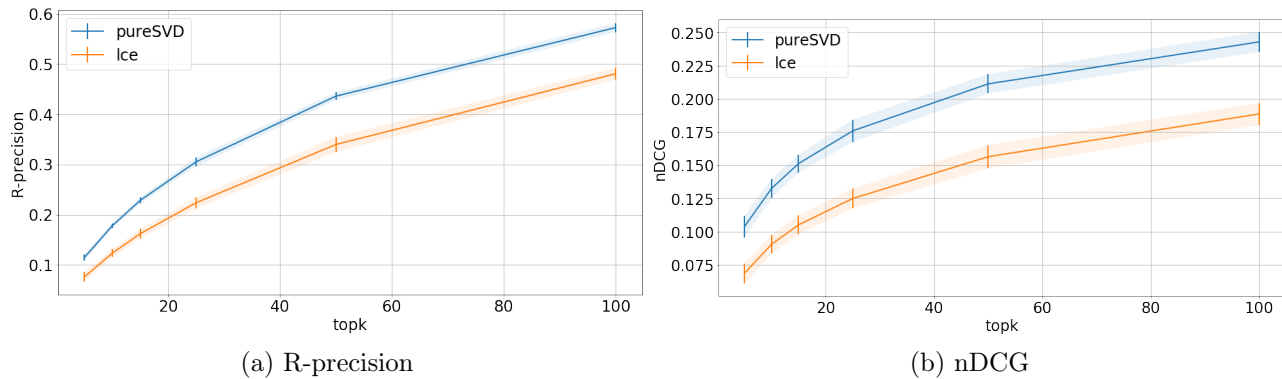


Рис. 1. Зависимость качества от topk.

Литература

- [1] Geoffray Bonnin, Dietmar Jannach. Automated Generation of Music Playlists: Survey and Experiments. ACM Computing Surveys (CSUR). 2014

- [2] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, Richard A. Harshman. Indexing by Latent Semantic Analysis. JASIS. 1990
- [3] Martin Saveski, Amin Mantrach. Item Cold-Start Recommendations: Learning Local Collective Embeddings. Proceeding RecSys '14 Proceedings of the 8th ACM Conference on Recommender systems Pages 89-96. 2014
- [4] Yifan Chen, Xiang Zhao. Leveraging High-Dimensional Side Information for Top-N Recommendation. CoRR. 2017
- [5] Cold-Start Item and User Recommendation with Decoupled Completion and Transduction Iman Barjasteh, Rana Forsati, Farzan Masrour, Abdol-Hossein Esfahanian, Hayder Radha. Cold-Start Item and User Recommendation with Decoupled Completion and Transduction. 2015