Tutorial

Recent Advances in Recommender Systems: Matrices, Bandits, and Blenders

Georgia Koutrika
Athena Research Center
Athens, Greece
georgia@imis.athena-innovation.gr

ABSTRACT

Recent years have witnessed an explosion in methods applied to solve the recommendation problem. Modern recommender systems have become increasingly more complex compared to their early content-based and collaborative filtering versions. In this tutorial, we will cover recent advances in recommendation methods, focusing on matrix factorization, multi-armed bandits, and methods for blending recommendations. We will also describe evaluation techniques, and outline open issues and challenges. The ultimate goal of this tutorial is to present a toolkit of new recommendation methods in perspective to data-related problems, and highlight opportunities and new research paths for researchers and practitioners that work on problems in the intersection of recommender systems and databases.

1 INTRODUCTION

The proliferation of digital content in a plurality of forms (including e-news, movies, and online courses), along with the popularity of portable devices has created immense opportunities as well as challenges for systems to provide users with information and services that best serve the users' needs. Matching consumers with the most appropriate items is key to enhancing user satisfaction and loyalty. Recommender systems come to the rescue providing advice on movies, products, travel, leisure activities, and many other topics. Personalized recommendations can elevate the user experience. That is why e-commerce leaders like Amazon and Netflix have made recommender technology a salient part of their systems [15].

Broadly speaking, recommender systems are based on one of two strategies. *Content-based filtering* creates a profile for each user or item to characterize their features. The profiles allow the recommender system to associate users with matching items. An alternative to content-based filtering relies only on past user behavior (e.g., previous purchases or user ratings). This approach is known as *collaborative filtering*, a term coined by the developers of Tapestry, an early recommender system [10]. Collaborative filtering analyzes relationships between users and interdependencies among items to identify new user-item associations based on which to make recommendations.

Recent years have witnessed an explosion in methods applied to solve the recommendation problem and modern recommender systems have become increasingly more complex. Matrix factorization methods, popularized with the Netflix prize [15], have become a dominant methodology within collaborative filtering recommenders due to their superior performance both in terms of recommendation quality and scalability. On the other hand, multi-armed bandits are becoming popular in interactive recommendation settings, for example for recommending songs in Pandora [29]. Ranked lists of items generated by different recommender systems are blended together into the final list of recommendations shown to the user. The blending problem is essentially a multi-objective optimization problem, with objectives such as relevancy, coverage and diversity competing with each other [6, 25]. Overall, the landscape of recommender systems has changed immensely since the first content-based and collaborative filtering systems emerged, and the state-of-the-art approaches show outstanding results and open up new opportunities and research paths.

Interestingly, we are used to thinking of recommendations in the context of systems that serve items, such as movies, products, friend connections, etc, to users. The reality is that the recommendation problem arises in many different scenarios beyond those targeting user consumption. For example, in the context of databases, such scenarios include but are not limited to data exploration, query optimization, visualization, data integration, and workflow design, where the purpose is to select tuples [7], queries [9], views [8], exploration actions [19], query plans [30], visualization graphs [26–28], work flows [12], and so forth. While there is work in these areas, it pales compared to the amount and diversity of recommendation methods developed for items such as movies and products.

Therefore, the purpose of this tutorial is two-fold. First, it aims at providing a comprehensive overview of recent advances in recommendation methods, highlighting their capabilities and their impact. The focus of the tutorial is on matrix factorization methods, multi-armed bandits, and blending methods. It will discuss major techniques, evaluation methodologies, and open issues. Since the recommendation problem appears in many different settings, it is the purpose of this tutorial to provide a solid framework for placing novel recommendation work into perspective for data-related problems, provide a toolkit of new methods, and highlight research opportunities for researchers and practitioners in database systems, data-intensive applications, and the intersection of recommender systems and databases.

The following sections describe the structure and contents of the tutorial. The tutorial does not require any prior knowledge in recommender systems since there will be detailed introductions to the relevant techniques.

2 OUTLINE

The tutorial is structured in the following parts.

2.1 Recommendation Framework

The objective of this section is to introduce the audience to the recommendation problem, define the basic concepts as well as the different instances of the problem (e.g., rating prediction and whole-page optimization), and provide an overview of the classical approaches for generating recommendations.

Recommender systems appeared back in the nineties, and two broad categories of recommendation approaches emerged: content-based and collaborative filtering. Content-based approaches analyze user past selections (e.g., web pages they visited, movies they watched) to learn user preferences and recommend items with similar content to the user's past selections and likes.

Collaborative filtering analyzes usage data (e.g., user ratings and purchases) and recommends to the user either items with similar usage characteristics as the items selected by this user or items from users with similar usage characteristics to this user.

We will explain the basic characteristics and operations behind each family of methods as well as their advantages and their shortcomings. The objective of this section is to lay the necessary foundations for the rest of the tutorial. It also aims at preparing the ground for understanding the methods to be presented subsequently and their impact.

2.2 Matrix Factorization

Matrix Factorization has gained popularity in recommender systems in recent years due to its superior performance both in terms of recommendation quality and scalability. Part of its success is due to the Netflix Prize contest for movie recommendations, which popularized a Singular Value Decomposition (SVD) based matrix factorization algorithm [13]. The Netflix Prize competition began in October 2006 and has fueled much recent progress in the field of collaborative filtering. For the first time, the research community gained access to a large-scale, industrial strength data set of 100 million movie ratings while the nature of the competition encouraged rapid development, where innovators built on each generation of techniques to improve prediction accuracy.

Experience with datasets such as the Netflix Prize data has shown that matrix factorization methods deliver accuracy superior to classical nearest-neighbor techniques [14]. At the same time, they offer a compact memory-efficient model that systems can learn relatively easily. What makes these techniques even more convenient is that models can integrate naturally many crucial aspects of the data, such as multiple forms of user feedback, temporal dynamics, and confidence levels.

In its basic form, matrix factorization characterizes both items and users by vectors of latent factors inferred from the ratings users gave to the items. Early systems use Singular Value Decomposition (SVD) – a well-established technique for identifying latent semantic factors in information retrieval – as a matrix factorization method for collaborative filtering. In the following years, several extensions to matrix factorization have been proposed and matrix factorization becomes the foundation in most recent recommender systems.

We will start with some background on Singular Value Decomposition and describe how early works (e.g., [23]) use SVD to capture latent relationships between customers and products and to produce a low-dimensional representation of the original customer-product space in order to compute the predicted likeliness of a certain product by a customer. We will introduce Low-rank Matrix Factorization [13, 21] followed by most recent important extensions to Matrix Factorization for recommendations, such as SLIM [20].

2.3 Multi–armed Bandits

Traditional recommender systems, including collaborative filtering, content-based filtering and hybrid approaches, can provide meaningful recommendations at an individual level by leveraging users' interests as demonstrated by their past activity. However, in many web-based scenarios (e.g., filtering news articles or display of advertisements), the content universe undergoes frequent changes, with content popularity changing over time as well. Furthermore, a significant number of visitors are likely to be entirely new with no historical consumption record whatsoever; this is known as a cold-start situation. These issues make traditional recommender approaches difficult to apply. In such highly dynamic recommendation domains, it is essential for the recommendation method to adapt to the shifting preference patterns of the users and the evolving space of items. Exploration-exploitation methods, a.k.a. multi-armed bandits, have been shown to be an excellent solution.

In probability theory, the multi-armed bandit problem is a problem in which a gambler at a row of slot machines (sometimes known as "one-armed bandits") has to decide which machines to play, how many times to play each machine and in which order to play them. When played, each machine provides a random reward from a probability distribution specific to that machine. The objective of the gambler is to maximize the sum of rewards earned through a sequence of lever pulls.

For example [17], in the context of article recommendation, we may view articles in the pool as arms. When a presented article is clicked, a reward of 1 is incurred; otherwise, the reward is 0. With this definition of reward, the expected reward of an article is precisely its clickthrough rate (CTR), and choosing an article with maximum CTR is equivalent to maximizing the expected number of clicks from users, which in turn is the same as maximizing the total expected reward in the bandit formulation. Furthermore, we may "summarize" users and articles by a set of informative features that describe them compactly. By doing so, a bandit algorithm can generalize CTR information from one article/user to another, and learn to choose good articles more quickly, especially for new users and articles.

We will first present context-free K-armed bandit algorithms [4, 22], such as ϵ-greedy and upper confidence-bound (UCB) algorithms [1, 3, 16], and then move to contextual bandit algorithms [17, 29]. We will focus on multi-armed bandits used in the context of recommender systems and in particular in three problems: (a) Popularity ranking, to balance exposure of new items (exploration) with old winners (exploitation), (b) Model-based collaborative filtering [11, 18], and (c) Dueling bandits, to efficiently compare multiple recommendation methods [5, 24].

2.4 Blending Models

Several domains require "blending" of recommendations from different sources. Blending allows different recommendation strategies to develop independently, and combine their outputs post-hoc into a meta-recommender. The result aims at providing recommendations of higher quality and diversity. For instance, the Pinterest Homefeed is a personalized feed of content (i.e., pins) drawn from many sources, including followed users, followed topics, and recommendations, among other sources. Each type of content is ranked by its own specialized machine learning model, and then blended with a ratio-based round-robin method to create the final Homefeed [6].

We will examine different methods to blend recommendations, including fixed ratio, greedy, calibrated ranker and multi-armed bandit-approaches [2, 6, 25]. As we examine these blending systems, new questions arise as to how to measure success. Unlike traditional search ranking problems, recommender systems face both short- and long-term optimization challenges as there is a need to balance immediate user-engagement metrics and long term ecosystem health. We will examine new such metrics and approaches to this end.

2.5 Lessons Learnt and Open Issues

In this section, we will discuss lessons learnt, open issues and new research directions created by these novel recommendation methods. We will also discuss about recommendation problems that do not target user consumption. In particular, we will describe such scenarios including data exploration, query optimization, visualization, data integration, workflow design, and so forth, where the purpose is to sort out not movies or products but queries, views, exploration actions, query plans, visualization graphs, and so forth. We will examine these problems in the light of the recent developments in the recommendation arena and discuss new research directions and opportunities that arise.

3 PRESENTER BIO

Georgia Koutrika is Director of Research at Athena Research Center in Greece. She has worked at HP Labs in Palo Alto, at IBM Research-Almaden in San Jose, and as a postdoctoral researcher at the Computer Science Dept., Stanford University. She has worked extensively on personalization and recommender systems, large-scale information extraction, entity resolution and information integration, and querying and data exploration interfaces. Her work has been incorporated in commercial products, has been described in 7 granted patents and 19 patent applications in the US and worldwide, and has been published in more than 80 research papers in top-tier conferences and journals. An IEEE Senior member, ACM member, and ACM SIGMOD Associate Information Director, Georgia has also served as a General Co-Chair for ACM SIGMOD 2016, Industrial Track PC Chair for EDBT 2016, and Workshop and Tutorial Co-Chair for IEEE ICDE 2016. She is currently Demo PC co-chair for ACM SIGMOD 2018.

REFERENCES

[1] Rajeev Agrawal. Sample mean based index policies with o(log n) regret for the multi-armed bandit problem. *Advances in Applied Probability*, 27(4):1054–1078, 1995.

[2] Amr Ahmed, Choon Hui Teo, S.V.N. Vishwanathan, and Alex Smola. Fair and balanced: Learning to present news stories. In *the Fifth ACM International Conference on Web Search and Data Mining*, WSDM '12, pages 333–342, New York, NY, USA, 2012. ACM.

[3] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002.

[4] Don Berry and Bert Fristedt. *Bandit Problems: Sequential Allocation of Experiments*, volume Monographs on Statistics and Applied Probability. Chapman and Hall, 1985.

[5] Bangrui Chen and Peter I. Frazier. Dueling bandits with weak regret. *CoRR*, abs/1706.04304, 2017.

[6] Stephanie deWet. Personalized content blending in the pinterest homefeed. In *2017 Netflix Workshop on Personalization, Recommendation and Search*, 2017.

[7] Marina Drosou and Evaggelia Pitoura. Ymaldb: exploring relational databases via result-driven recommendations. *The VLDB Journal*, 22(6):849–874, Dec 2013.

[8] Humaira Ehsan, Mohamed A. Sharaf, and Panos K. Chrysanthis. Muve: Efficient multi-objective view recommendation for visual data exploration. In *the 32nd IEEE International Conference on Data Engineering, ICDE 2016, Helsinki, Finland, May 16-20, 2016*, pages 731–742, 2016.

[9] Magdalini Eirinaki, Suju Abraham, Neoklis Polyzotis, and Naushin Shaikh. Querie: Collaborative database exploration. *IEEE Trans. Knowl. Data Eng.*, 26(7):1778–1790, 2014.

[10] David Goldberg, David A. Nichols, Brian M. Oki, and Douglas B. Terry. Using collaborative filtering to weave an information tapestry. *Commun. ACM*, 35(12):61–70, 1992.

[11] Frédéric Guillou, Romaric Gaudel, and Philippe Preux. Collaborative filtering as a multi-armed bandit. In *the Workshop: Machine Learning for eCommerce*, NIPS'15, 2015.

[12] Dietmar Jannach, Michael Jugovac, and Lukas Lerche. Supporting the design of machine learning workflows with a recommendation system. *TiiS*, 6(1):8:1–8:35, 2016.

[13] Yehuda Koren. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In *the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 426–434, New York, NY, USA, 2008. ACM.

[14] Yehuda Koren and Robert Bell. *Advances in Collaborative Filtering*, pages 145–186. Springer US, Boston, MA, 2011.

[15] Yehuda Koren, Robert M. Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.

[16] T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.

[17] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *the 19th International Conference on World Wide Web*, WWW '10, pages 661–670, New York, NY, USA, 2010. ACM.

[18] Shuai Li, Claudio Gentile, Alexandros Karatzoglou, and Giovanni Zappella. Data-dependent clustering in exploration-exploitation algorithms. *CoRR*, abs/1502.03473, 2015.

[19] Tova Milo and Amit Somech. React: Context-sensitive recommendations for data analysis. In *the 2016 International Conference on Management of Data*, SIGMOD '16, pages 2137–2140, New York, NY, USA, 2016. ACM.

[20] Xia Ning and George Karypis. Slim: Sparse linear methods for top-n recommender systems. In *ICDM*, 2011.

[21] Arkadiusz Paterek. Improving regularized singular value decomposition for collaborative filtering. *KDD Cup and Workshop*, 2007.

[22] Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

[23] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John T. Riedl. Application of dimensionality reduction in recommender system - a case study. In *the KDD Workshop on Web Mining for e-Commerce: Challenges and Opportunities (WebKDD)*, 2000.

[24] Yanan Sui, Vincent Zhuang, Joel W. Burdick, and Yisong Yue. Multi-dueling bandits with dependent arms. *CoRR*, abs/1705.00253, 2017.

[25] Choon Hui Teo, Houssam Nassif, Daniel Hill, Sriram Srinivasan, Mitchell Goodman, Vijai Mohan, and S.V.N. Vishwanathan. Adaptive, personalized diversity for visual discovery. In *the 10th ACM Conference on Recommender Systems*, RecSys '16, pages 35–38, New York, NY, USA, 2016. ACM.

[26] Manasi Vartak, Silu Huang, Tarique Siddiqui, Samuel Madden, and Aditya G. Parameswaran. Towards visualization recommendation systems. *SIGMOD Record*, 45(4):34–39, 2016.

[27] Manasi Vartak, Sajjadur Rahman, Samuel Madden, Aditya G. Parameswaran, and Neoklis Polyzotis. SEEDB: efficient data-driven visualization recommendations to support visual analytics. *PVLDB*, 8(13):2182–2193, 2015.

[28] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. Towards a general-purpose query language for visualization recommendation. In *Workshop on Human-In-the-Loop Data Analytics*, HILDA '16, pages 4:1–4:6, New York, NY, USA, 2016. ACM.

[29] Tao Ye and Mohit Singh. Combining relevancy scoring and contextual bandits for personalized ranking in music discovery. In *the 2017 Netflix Workshop on Personalization, Recommendation and Search*, 2017.

[30] Jihad Zahir and Abderrahim El Qadi. A recommendation system for execution plans using machine learning. *Mathematical and Computational Applications*, 21(2), 2016.