

## ОБЗОР СОВРЕМЕННЫХ РЕКОМЕНДАТЕЛЬНЫХ СИСТЕМ

## MODERN RECOMMENDER SYSTEMS' REVIEW

K. Nikolaev

*Summary.* Nowadays users must sort through large amount of diverse information to find the most relevant one. Recommender systems improve the efficiency of this process; they deal with huge number of items and offer content portion that users would be interested in. The paper describes one of the most popular recommender systems used for production purposes and of great practical importance in industry. This work aims to review the existing research and analyze models' features presented there. The contributions of the paper are the following: analysis of mentioned recommender systems, describing their application areas, strengths, and weaknesses. The conclusion focuses on the practical value of explorations and opportunities for other industries.

*Keywords:* recommender systems, recurrent neural networks, reinforcement learning, cold start, machine learning.

Николаев Константин Сергеевич

Ассистент, Национальный исследовательский университет «МИЭТ»  
knpreacher@gmail.com

*Аннотация.* В современном мире человек взаимодействует с большим количеством разнообразной информации с целью поиска наиболее актуальной для него. Для улучшения эффективности этого взаимодействия используются различные рекомендательные системы, обрабатывающие многомиллионные массивы элементов и предлагающие пользователю наиболее релевантные. В данной статье представлены одни из самых популярных рекомендательных систем, которые развернуты в производственных масштабах и показали свою практическую значимость в реальных условиях. Целью работы является обзор ряда существующих исследований и анализ функционала представленных в них моделей. В статье решаются следующие задачи: обзор ряда рекомендательных систем, описание областей их применения, выделение основных преимуществ и недостатков каждой модели. В результате сделан вывод об актуальности представленных рекомендательных систем и их пригодности в контексте использования в других отраслях.

*Ключевые слова:* рекомендательная система, рекуррентные нейронные сети, обучение с подкреплением, холодный старт, машинное обучение.

## Введение

В настоящее время повсеместно используются рекомендательные системы. Они применяются в различных предметных областях и зачастую нацелены на улучшение взаимодействия пользователя с платформой, будь то интернет-магазин, веб-ресурс для просмотра видео или музыкальный сервис. Рекомендательные системы решают задачи обработки большого массива разнообразных данных для того, чтобы определить предпочтения пользователя и своевременно предлагать ему интересующий его контент, адаптируясь к изменениям в его поведении. Учитывая вышеизложенное, основными критериями при разработке таких систем является их быстродействие, степень персонализации выдаваемых рекомендаций и скорость перестройки под меняющиеся предпочтения, что в итоге напрямую влияет на удовлетворенность пользователя.

В рамках данной работы проанализированы две наиболее известные рекомендательные системы: разработка компании Google [1], основанная на так называемом REINFORCE [8], и разработка компании Amazon Web Services [3] — иерархическая рекуррентная нейронная сеть, использующая метаданные. Для каждой из них рассмотрена их практическая значимость, подходы к проектированию, методы устранения потенциальных проблем и способы повышения их эффективности. В заключении оценивается применимость данных разработок в других

областях деятельности, в частности, в сфере лазерных технологий.

## Существующие подходы к проектированию рекомендательных систем

## 1. Основанная на REINFORCE модель

## 1.1. Описание применения

В исследовании рассматриваются классические модели обучения с подкреплением. В результате их анализа было выявлено следующее:

- существующие алгоритмы непригодны для постоянно разрастающегося пространства действий и состояний, обусловленного изменением предпочтений пользователей;
- подходы самовоспроизведения и моделирования [5] для устранения ограниченности данных неприменимы для динамической рекомендательной системы;
- большинство алгоритмов рекомендуют только один элемент, в то время как реальные системы одновременно предлагают сразу несколько [7].

В предложенном исследовании все эти особенности учтены при разработке, в результате разработана «Тор-К» корректировка «вне стратегии». Эта модель эффективно используется на платформе YouTube для предложения пользователям актуальных для них видео.

1.2. Архитектура модели

1.2.1. Моделирование

Задача состоит в том, чтобы каждый последующий рекомендованный элемент увеличивал показатель удовлетворенности пользователя. Она преобразуется в марковский процесс принятия решений  $(S, A, P, R, \rho_0, \gamma)$ , где

- $S$ : непрерывное пространство состояний;
- $A$ : дискретное пространство действий;
- $P : S \times A \times S \rightarrow \mathbb{R}$ : функция переходов;
- $R : S \times A \rightarrow \mathbb{R}$ : функция награды;
- $\rho_0$ : распределение начального состояния;
- $\gamma$ : коэффициент дисконтирования.

В качестве архитектуры рекуррентной нейронной сети была выбрана Chaos Free RNN [2] за счет ее стабильности и вычислительной эффективности. Сама стратегия моделируется следующим образом:

$$\pi_\theta(a|s) = \frac{\exp(s^T v_a / T)}{\sum_{a' \in A} \exp(s^T v_{a'} / T)} \quad (1)$$

где  $v_a \in \mathbb{R}^n$  дополнительное вложение для каждого действия  $a$  в пространстве действий  $A$ , а  $T$  — коэффициент, обычно 1.

1.2.2. Методы уменьшения погрешности

Для снижения дисперсии градиента ожидаемой суммарной награды применяется коэффициент дисконтирования  $\gamma$ . В качестве дополнительных методов ее уменьшения были рассмотрены следующие:

- ограничение веса;
- нормализованная выборка важности;
- оптимизация стратегии доверенного региона (TRPO) [4].

При проведении экспериментов по результатам регрессионных тестов наиболее эффективным оказалось ограничение веса, поэтому в дальнейшем именно этот метод был использован в итоговой модели.

1.2.3. Адаптация REINFORCE для выбора  $K$  элементов

Чтобы адаптировать REINFORCE к предложенной модели, использовано скорректированное значение градиента, тогда коэффициент поправки на отклонение от стратегии равен:

$$\sum_{s_t \sim d_t^{\pi_\theta(\cdot|s_t)}, a_t \sim \beta(\cdot|s_t)} \left[ \frac{\pi_\theta(a_t|s_t)}{\beta(a_t|s_t)} \frac{\partial \alpha(a_t|s_t)}{\partial \pi(a_t|s_t)} R_t(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t|s_t) \right] \quad (2)$$

Особое значение здесь имеет дополнительный множитель к основному поправочному коэффициенту вне стратегии  $\frac{\pi(a|s)}{\beta(a|s)}$ :

$$\lambda_K(s_t, a_t) = \frac{\partial \alpha(a_t|s_t)}{\partial \pi(a_t|s_t)} = K(1 - \pi_\theta(a_t|s_t))^{K-1} \quad (3)$$

Пока искомый элемент не выбран, стратегия с каждым шагом увеличивает его вероятность. Как только она достигает заданного порога, он попадает в набор для рекомендации и его вероятность больше не увеличивается, что позволяет отбирать другие релевантные элементы.

1.3. Преимущества

Представленный в исследовании алгоритм характеризуется следующей особенностью: он учитывает как выбранные пользователем рекомендации, так и те, что были им проигнорированы. Также он ориентирован на одновременную выдачу нескольких подходящих элементов так, чтобы получаемая за них суммарная награда была максимальной. Совокупность этих преимуществ позволяет говорить об эффективности разработки, что подтверждается повышением удовлетворенности пользователей.

1.4. Недостатки

Исходя из представленного описания разработки, можно сделать вывод о том, что она обладает рядом недостатков. Одним из них является ограниченность данных: если пользователи мало взаимодействовали с объектами, нейронная сеть плохо обучится, что негативно скажется на качестве рекомендаций. Кроме того, из-за сложностей в оценке вероятностей на основе исторических данных у выдаваемых рекомендательной системой предсказаний может быть низкая точность. Еще одним недостатком можно считать необходимость тщательной настройки гиперпараметров, что требует определенного времени и экспертных навыков. Также алгоритмы подобного рода требуют значительных вычислительных ресурсов, что может повлиять на возможность их масштабирования и эффективность в реальном времени. Следовательно, при внедрении данной технологии стоит учитывать ее возможные проблемы и продумать пути их решения.

Вывод

Рекомендательная система, основанная на алгоритме REINFORCE, хорошо показала себя при производственных испытаниях в реальных условиях (рекомендация видео на YouTube). Несмотря на ряд недостатков, она повысила удовлетворенность пользователей и увеличила время и количество просмотра видео. Таким образом, она оказалась особенно эффективной для рекомендаций видеоконтента.

2. Динамическая модель контекстных рекомендаций

2.1. Описание применения

Следующее исследование посвящено разработке рекомендательной системы (так называемой ИРНС-мета),

способной приспосабливаться к изменяющимся предпочтениям пользователя и обучаться на истории его взаимодействия с элементами; она решает проблему холодного старта, использует исторические данные для прогнозирования и масштабируется для многомиллионных массивов элементов.

Эффективность разработки демонстрируется в ходе реальных и искусственных экспериментов, а отдельные компоненты развернуты в производственных масштабах и служат в качестве базового рекомендательного движка для тысяч веб-сайтов. В исследовании упоминаются следующие области применения данной технологии: рекомендация фильмов по истории просмотра (Netflix) и по метаданным при холодном старте (MovieLens), подборка тематических сообществ по интересующим пользователя темам и его активности (Reddit), определение наиболее релевантных и актуальных новостей с учетом их быстрого устаревания (Outbrain) и прочие сферы, где требуется высокая степень персонализации в постоянно изменяющихся условиях (Yoochoose, Taobao).

## 2.2. Архитектура модели

Персонализированные рекомендации основаны на истории взаимодействия, где каждая строка — это запись вида {(время, идентификатор пользователя, идентификатор элемента, идентификатор действия)} =  $\{(t_k, u_k, a_k, v_k) : k = 1, \dots, n\}$ .

Один из компонентов ИРНС-мета — последовательные модели. Действия пользователей сгруппированы в упорядоченные наборы  $x_i = [x_{i1}, x_{i2}, \dots, x_{ik}]$  так, что  $x_{ik} = (a_{ik}, t_{ik}, v_{ik})$  и  $t_{ik} > t_{ij} | k > j$ . Пусть  $A_{k+1}$  — случайная переменная, представляющая следующий элемент  $a_{k+1}$ ,  $x_{1:k} = (x_1, \dots, x_k)$  — история взаимодействия, а  $\phi = (\phi_1, \dots, \phi_m)^T$  — векторная функция оценки для каждого из  $m$  элементов, тогда:

$$a_{k+1} \sim p(A_{k+1} | x_{1:k}) = \text{softmax}(\phi(h(x_{1:k}))), \quad (4)$$

где  $h_k = h(x_{1:k})$  — полученное в результате обучения представление соответствующей истории взаимодействия. (4) описывает предсказание одного элемента с учетом всей истории пользователя, на практике же для всех событий одновременно применяется обратное распространение во времени.

В иерархических рекуррентных нейронных сетях используются так называемые сеансы, в пределах которых интересы пользователя считаются постоянными. Нижние слои сети отслеживают динамику в пределах одного сеанса, в то время как верхние обучаются на межсеансовых данных. При объединении слоев возможна та-

кая ситуация, когда верхние слои не могут обновиться, так используются более чем одним пользователем. Для устранения этой проблемы собираемые от пользователя отклики объединяются с контекстной информацией (язык запроса, цена товара и пр.).

Также в ИРНС-мета используются свойства элементов. Пусть  $f_j$  — вектор свойств  $j$ -го элемента,  $w_j = w_j(f_j)$  — его отображение,  $b = b_j(f_j)$  — смещение. Тогда дешифратор можно представить как модель взаимодействия второго порядка между  $w_j$  и  $h_k$  и объединить оба представления следующим образом:

$$\tilde{\phi}_j(h_k) = (1 - \lambda)(w_j^T h_k + b_j) + \lambda(w(f_j)^T h_k + b(f_j)), \quad (5)$$

где  $w(\cdot), b(\cdot)$  — обучаемые функции, в которые встраиваются свойства элементов, а  $0 \leq \lambda \leq 1$  — коэффициент смешения, предотвращающий повторное обучение [6] при произвольном значении и использующийся в холодном старте на метаданных при значении, равном 1.

## 2.3. Преимущества

В отличие от многих популярных рекомендательных систем, ИРНС-мета может хорошо персонализировать рекомендации, основываясь только на данных последнего элемента, с которым взаимодействовал пользователь. Кроме того, за счет кодирования временных меток она показывает результаты, сопоставимые с результатами других моделей, при значительно меньших вычислительных затратах. Еще одной особенностью данной разработки является внедрение метаданных при холодном старте: в ходе экспериментов на реальных датасетах было продемонстрировано, что такой подход существенно повышает качество рекомендаций. Также важным преимуществом ИРНС-мета является короткое время обучения (<6 часов), обусловленное использованием выборки по важности, дающей результаты, близкие к эталонным.

## 2.4. Недостатки

Архитектура ИРНС-мета подразумевает работу с сеансами; по этой причине она обладает следующим недостатком: при выдаче рекомендаций на основе данных предыдущих сеансов значительно снижается эффект персонализации, так как модель «забывает» предпочтения пользователя. Более того, на рекомендательную систему одновременно влияет множество источников информации, поэтому сбой в любом ее компоненте может критически сказаться на работе всей системы. Также стоит отметить, что поведение модели изучено не в полной мере, так как ее обучение происходило на открытых датасетах, обладающих рядом ограничений и не позволяющих тщательно исследовать взаимодействие между различными факторами, важными для динамичной среды.

## Вывод

Исходя из проведенного анализа ИРНС-мета, можно сделать вывод о том, что она хорошо адаптируется к быстро изменяющимся предпочтениям пользователя и эффективно решает проблему холодного старта за счет использования метаданных, что делает эту рекомендательную систему вполне конкурентоспособной в настоящее время. При этом стоит учитывать некоторые ее особенности (например, сеансы), и применять ее только в тех сферах, в которых эти факторы не будут оказывать существенного влияния на ее качество и производительность.

## Заключение

В статье рассмотрены одни из самых популярных рекомендательных систем, использующиеся в промышленных масштабах. Они эффективно решают задачи, связанные с выявлением пользовательских предпочтений и выдачей наиболее релевантных рекомендаций. Такая

ориентация на интересы пользователя делает эти рекомендательные системы хорошим решением в тех сферах деятельности, которые ориентированы на потребление материальных товаров или информационных ресурсов: социальные сети, видеохостинги, интернет-магазины, музыкальные платформы, различные веб-сайты. Тем не менее, они не являются универсальными. Так, в некоторых предметных областях объект рекомендации не может быть представлен в виде единичной записи с уникальным идентификатором, так как включает в себя множество компонентов, и смысловую нагрузку несет именно их комбинация. В частности, в сфере лазерных технологий базовым объектом является набор параметров и их значений. Такие данные в большом количестве существуют в лазерной промышленности и набираются в результате проведения многочисленных экспериментов. Рассмотренные в статье модели не приспособлены к обработке таких структур, поэтому проблема подбора подходящего набора параметров для лазерной промышленности остается открытой.

## ЛИТЕРАТУРА

1. Chen M., Beutel A., Covington P., Jain S., Belletti F., Chi E. Top-K Off-Policy Correction for a REINFORCE Recommender System // WSDM '19. 2019. DOI: 10.1145/3289600.3290999.
2. Laurent T., Brecht J. A recurrent neural network without chaos // ICLR. 2017. DOI: 10.48550/arXiv.1612.06212.
3. Ma Y., Narayanaswamy B., Lin H., Ding H. Temporal-Contextual Recommendation in Real-Time // KDD '20. 2020. DOI: 10.1145/3394486.3403278
4. Schulman J., Levine S., Abbeel P., Jordan M., Moritz P. Trust region policy optimization. // International Conference on Machine Learning. 2015, С. 1889–1897. DOI: 10.48550/arXiv.1502.05477.
5. Silver D., Huang A., Maddison C. et al Mastering the game of Go with deep neural networks and tree search // Nature. 2016, Т. 529, №7587, С. 484–489. DOI: 10.1038/nature16961.
6. Srivastava N., Hinton G., Krizhevsky A., Sutskever I., Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting // The Journal of Machine Learning Research. 2014, №15(1), С. 1929–1958. DOI: 10.5555/2627435.2670313.
7. Swaminathan A., Krishnamurthy A., Agarwal A., Dudik M., Langford J., Jose D., Zitouni I. Off-policy evaluation for slate recommendation // Advances in Neural Information Processing Systems. 2017. DOI: 10.48550/arXiv.1605.04812.
8. Williams R.J. Simple statistical gradient-following algorithms for connectionist reinforcement learning // Machine Learning. 1992, Т. 8, № 3–4, С. 229–256. DOI: 10.1007/BF00992696.

© Николаев Константин Сергеевич (knpreacher@gmail.com)

Журнал «Современная наука: актуальные проблемы теории и практики»