

# СРАВНИТЕЛЬНЫЙ АНАЛИЗ РЕКОМЕНДАТЕЛЬНЫХ АЛГОРИТМОВ ДЛЯ ОПРЕДЕЛЕНИЯ ПЕРСОНАЛЬНЫХ ПРЕДПОЧТЕНИЙ НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ

**Ян Тяньци**

Московский государственный технический университет им. Н.Э. Баумана (национальный исследовательский университет)  
2936703080@qq.com

**Афанасьев Геннадий Иванович**

К.т.н., доцент, Московский государственный технический университет им. Н.Э. Баумана (национальный исследовательский университет)  
gaipcs@bmstu.ru

**Калистратов Алексей Павлович**

Ассистент, Московский государственный технический университет им. Н.Э. Баумана (национальный исследовательский университет)  
akalistratov@gmail.com

**Афанасьев Арсений Геннадьевич**

Ассистент, Московский государственный технический университет им. Н.Э. Баумана (национальный исследовательский университет)  
afanasievag@bmstu.ru

## COMPARATIVE ANALYSIS OF RECOMMENDATION ALGORITHMS FOR DETERMINING PERSONAL PREFERENCES BASED ON NEURAL NETWORKS

**T. Yang  
G. Afanasyev  
A. Kalistratov  
A. Afanasyev**

*Summary:* In this paper, we study several popular recommendation algorithms. Among the many personalized recommendation algorithms, the collaborative filtering algorithm has become one of the most widely used technologies due to its simplicity, efficiency, and accuracy. Based on existing research, this article combines Slope One's advanced weighted method with deep learning autoencoder to learn deeper features of the dataset while alleviating the problem of data sparseness, thereby improving the quality of recommendations.

*Keywords:* personalized recommendation, scoring matrix, deep learning autoencoder, neural networks.

*Аннотация.* В данной работе производится исследование нескольких популярных алгоритмов рекомендаций. Среди многих алгоритмов персонализированных рекомендаций алгоритм совместной фильтрации стал одной из наиболее широко используемых технологий благодаря своей простоте, эффективности и точности. Основываясь на существующих исследованиях, в этой работе усовершенствованный взвешенный метод Slope One сочетается с автокодировщиком, основанном на глубоком обучении, чтобы изучить более глубокие особенности набора данных, одновременно облегчая проблему разреженности данных, тем самым улучшая качество рекомендаций.

*Ключевые слова:* персонализированная рекомендация, скоринговая матрица, автоэнкодер на основе глубокого обучения, нейронные сети.

## Введение

В настоящее время (по данным 2022 года) рынок онлайн-кинотеатров превышает 1 млрд долларов в РФ, а оборот главного интернет-гиганта мира, Netflix, составляет 32 млрд долларов. Потребители онлайн-кинотеатров часто следуют рекламе или рекомендациям, а значит, алгоритм рекомендаций является наиболее важной частью системы рекомендаций фильмов [1-6]. Однако, с резким увеличением количества пользователей и фильмов качество рекомендаций традиционных моделей серьезно ухудшается. Кроме того, традиционный алгоритм коллаборативной фильтрации не может эффективно извлекать нелинейные факторы в данных и рассматривает данные с одной точки зрения [7].

Проведение анализа настроений — это нечто большее, чем классификация документа или предложения по положительным или отрицательным классам. Действительно, поиск настроения, обсуждаемого в каждом аспекте или особенности объекта, имеет первостепенное значение. Поэтому, в зависимости от требуемой степени детализации, задача анализа настроений выполняется на уровне документа, предложения и аспекта.

В конце 2002 года концепция анализа настроений, которая в основном фокусируется на выявлении полярности всего документа, содержащего мнения, по отношению к единому объекту (например, классификация всего обзорного документа, в котором говорится о данной футбольке). Однако анализ настроений на уровне документа позже оказался ограниченным для предостав-

ления достаточного количества информации, поскольку он не учитывает различные предложения и аспекты, которые могут содержаться в документе. Например, предложение «Это очень хороший фильм, который я посмотрел!» классифицируется как выражающее позитивное настроение. Однако анализ настроений на уровне предложения сталкивается с той же проблемой, что и анализ настроений на уровне документа, поскольку не предоставляет достаточной информации о том, что на самом деле нравится или не нравится клиенту, поскольку предложение может содержать несколько объектов с разными аспектами.

Следовательно, чтобы решить проблему работы с несколькими объектами с различными аспектами, ученые начали новую веху анализа настроений на уровне характеристик, внедрив методику сбора мнений на основе характеристик и обобщения, которая учитывает различные аспекты продукта и полярность их настроений. В настоящее время анализ настроений на этом уровне широко известен как аспектный анализ настроений. В частности, он анализирует различные характеристики объекта и точно определяет, что кому-то нравится или не нравится в этом объекте или аспекте объекта в обсуждении. Рассматривая следующий пример: «Еда хорошая, но обслуживание плохое», мы можем ясно видеть, что предложение является положительным по отношению к аспекту «еда», но отрицательным по отношению к аспекту «обслуживание». Таким образом, аспектный анализ является одной из самых сложных, но крайне необходимых задач анализа настроений. Более того, сегодняшние приложения анализа настроений в реальной жизни основаны на этом уровне. Таким образом, аспектный анализ включает в себя множество подзадач, включая извлечение аспектных терминов, анализ настроений аспектных терминов и анализ настроений аспектных категорий.

С развитием технологии глубокого обучения новой тенденцией развития стала интеграция традиционных алгоритмов рекомендаций с технологией глубокого обучения для решения вышеуказанных проблем. Цель данной работы — рассмотреть метод заполнения матрицы скоринга на основе улучшенного алгоритма SlopeOne, чтобы предложить идеи для смягчения проблемы разреженности данных.

**Алгоритм самокодирования рекомендаций**

Алгоритм Slope One — это алгоритм совместной фильтрации на основе элементов, предложенный профессором Даниэлем Лемиром в 2005 году, который генерирует прогнозы путем вычисления отклонений в оценках между элементами. По сравнению с другими алгоритмами коллаборативной фильтрации, алгоритм Slope One несколько точнее [8].

Наши алгоритмы Slope One работают на интуитивно понятном принципе «разницы в популярности» между элементами для пользователей. Парноно мы определяем, насколько один товар нравится больше другого. Один из способов измерить эту разницу — просто вычесть среднюю оценку двух пунктов.

В свою очередь, эта разница может быть использована для прогнозирования оценки другим пользователем одного из этих элементов, учитывая их оценку другого. Рассмотрим двух пользователей *A* и *B*, два элемента *I* и *J* и рис. 1. Пользователь *A* присвоил элементу *I* оценку 1, в то время как пользователь *B* присвоил ему оценку 2, в то время как пользователь *A* присвоил элементу *J* оценку 1,5. Мы видим, что элемент *J* оценивается выше, чем элемент *I*, на  $1,5 - 1 = 0,5$  балла, таким образом, мы могли бы предсказать, что пользователь *B* присвоит элементу *J* оценку  $2 + 0,5 = 2,5$ . Мы называем пользователя *B* ожидаемым пользователем и элемент *J* — ожидаемым элементом. В обучающем наборе существует множество таких различий для каждого неизвестного рейтинга, и мы берем среднее значение этих различий. Семейство схем первого уклона, представленных здесь, основано на трех способах выбора соответствующих дифференциалов для получения единого прогноза.

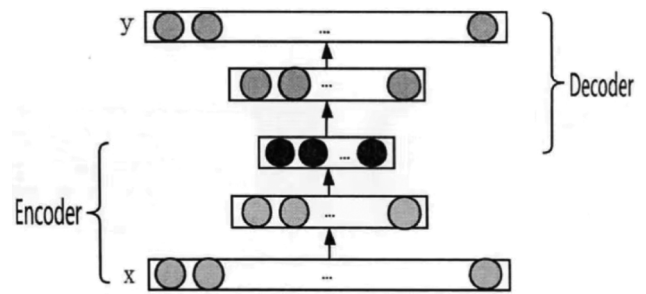


Рис. 1. Структура сети для алгоритмов рекомендаций с глубоким самокодированием

Рассчитаем коэффициент корреляции Пирсона (Pearson Correlation Coefficient) для расчета сходства между элементами, и его расчет основан на общей оценке двух элементов.

$$p_{ij} = \frac{\sum_{u \in U} (r_{ui} - \bar{r}_i)(r_{uj} - \bar{r}_j)}{\sqrt{\sum_{u \in U} (r_{ui} - \bar{r}_i)^2} \sqrt{\sum_{u \in U} (r_{uj} - \bar{r}_j)^2}} \quad (1)$$

где  $r_{ui}$  — Оценка товара *i* пользователем *u*;  $p_{ij}$  — обозначает сходство по Пирсону между товаром *i* и товаром *j*; *U* — множество всех пользователей;  $\bar{r}_i$  — Среднее значение баллов по пункту *i*;  $p_{ij} > 0$  — Пункт *i* и пункт *j* положительно коррелируют между собой;  $p_{ij} = 0$  — Пункт *i* и пункт *j* не связаны между собой;  $p_{ij} < 0$  — Пункт *i* и пункт *j* отрицательно коррелируют.

Поскольку некоррелированные и отрицательные корреляции не являются значимыми для прогнозиро-

вания баллов, в расчеты были включены только случаи, когда  $p_{ij} > 0$ . Таким образом, взвешенный алгоритм SlopeOne после слияния сходств элементов предсказывает формулу оценки  $pre_{ui}$  следующим образом.

$$pre_{ui} = \frac{\sum_{j \in S_{(u)} - \{i\}} (r_{uj} + dev_{ij}) * p_{ij} * |S_{ij}|}{\sum_{j \in S_{(u)} - \{i\}} * p_{ij} * |S_{ij}|} \quad (2)$$

где  $S_{ij}$  — Набор пользователей, которые оценили как элемент  $i$ , так и элемент  $j$ ;  $|S_{ij}|$  — Количество пользователей в наборе  $S_{ij}$ ;  $S_{(u)}$  — Коллекция предметов, оцененных пользователем  $u$ ;  $S_{(u)} - \{i\}$  — Набор элементов, у которых хотя бы один элемент  $j$  ( $i \neq j$ ) был оценен пользователем  $u$  одновременно с элементом  $i$ .

Поскольку количество пользователей обычно намного больше, чем количество фильмов, в практическом применении рекомендации фильмов, рекомендация на основе пользователя больше соответствует реальным потребностям. На основе алгоритма рекомендации U-AutoRec структура самокодера, содержащая только один скрытый слой, модифицируется и расширяется в глубокий самокодер, содержащий три скрытых слоя, и используется в задаче предсказания рейтинга фильма [9].

На рисунке 1 показана сетевая структура предложенного алгоритма глубокого самокодирования рекомендаций с тремя скрытыми слоями. Первые два слоя — это слои кодирования,  $e_1$  и  $e_2$  соответственно, которые выполняют функцию кодирования, т.е.  $encode(x) : R^n \rightarrow R^d$ , а вторые два слоя — это слои декодирования,  $d_1$  и  $d_2$  соответственно, которые выполняют функцию декодирования, т.е.  $decoder(x) : R^d \rightarrow R^n$ . Когда  $x$  является входным сигналом, целью самокодирования является минимизация разрыва между входным сигналом  $x$  и выходным сигналом  $y = decoder(encode(x))$ . Вся структура сети является полносвязной, где  $f$  — нелинейная функция активации [10].

На основе алгоритма глубокого самокодирования рекомендаций строится самокодирование с уменьшением шума путем искусственного «повреждения» исходных данных рейтинга фильмов на локальном уровне, и, соответственно, предлагается алгоритм глубокого самокодирования с уменьшением шума на основе самокодирования

рекомендаций фильмов под названием DeerRec. Это распространенная техника решения проблемы недостающих значений путем стирания определенных данных или замены некоторых из них [11]. Процесс добавления маскирующего шума показан на рисунке 2 ниже.

Как видно на рисунке 2, исходными входными данными являются  $x$ , а после маскировки и добавления шума данные становятся метаданными, причем некоторые элементы данных устанавливаются в 0. После этого испорченные данные подаются в самокодер для вычисления  $u$  и конечного выхода  $z$ . Оптимизируя функцию потерь  $LH(x, z)$  для исходного входа  $x$  и входа  $z$ , самокодер вынужден изучать особенности данных на основе «испорченных» данных. Самокодировщик вынужден изучать особенности данных на основе «испорченных» данных [7].

### Самокодеры с шумоподавлением

Самокодеры с шумоподавлением — это класс самокодеров, которые принимают на вход «испорченные» данные и обучаются предсказывать исходные, не испорченные данные на выходе [12]. На основе самокодера для подавления шума, предлагаемый глубокий самокодер для подавления шума формируется путем увеличения количества скрытых слоев, которые могут вычислять более сложную информацию о данных и изучать особенности высшего порядка входных данных. Как глубокая сеть с многослойной структурой, каждый уровень глубокого шумоподавляющего самокодера соответствует каждому слою нейронной сети, где выход предыдущего скрытого слоя используется в качестве входа для следующего слоя и так далее.

Как показано на рисунке 3, полный процесс обучения глубокого шумопонижающего самокодера достигается путем выполнения сначала предварительного обучения без контроля на каждом отдельном скрытом слое блока шумопонижающего самокодера, затем стекирования и, наконец, общего обучения с обратной настройкой.

На основе содержимого рисунка 3 строится самокодер глубокого шумоподавления, и модель обучается с новой матрицей оценок в качестве входных данных для минимизации функции потерь. Вход алгоритма: Матрица рейтинга пользователя — элемента  $R^{m \times n}$ , матрица

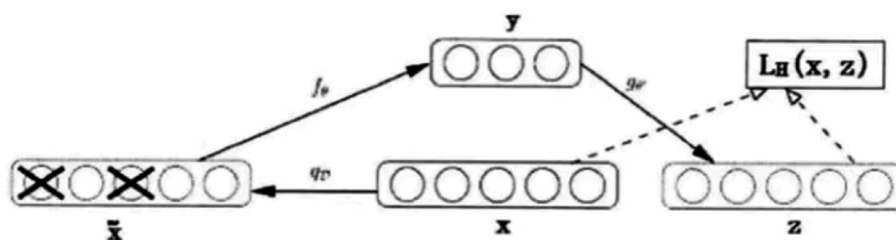


Рис. 2. Добавление маскирующего шума

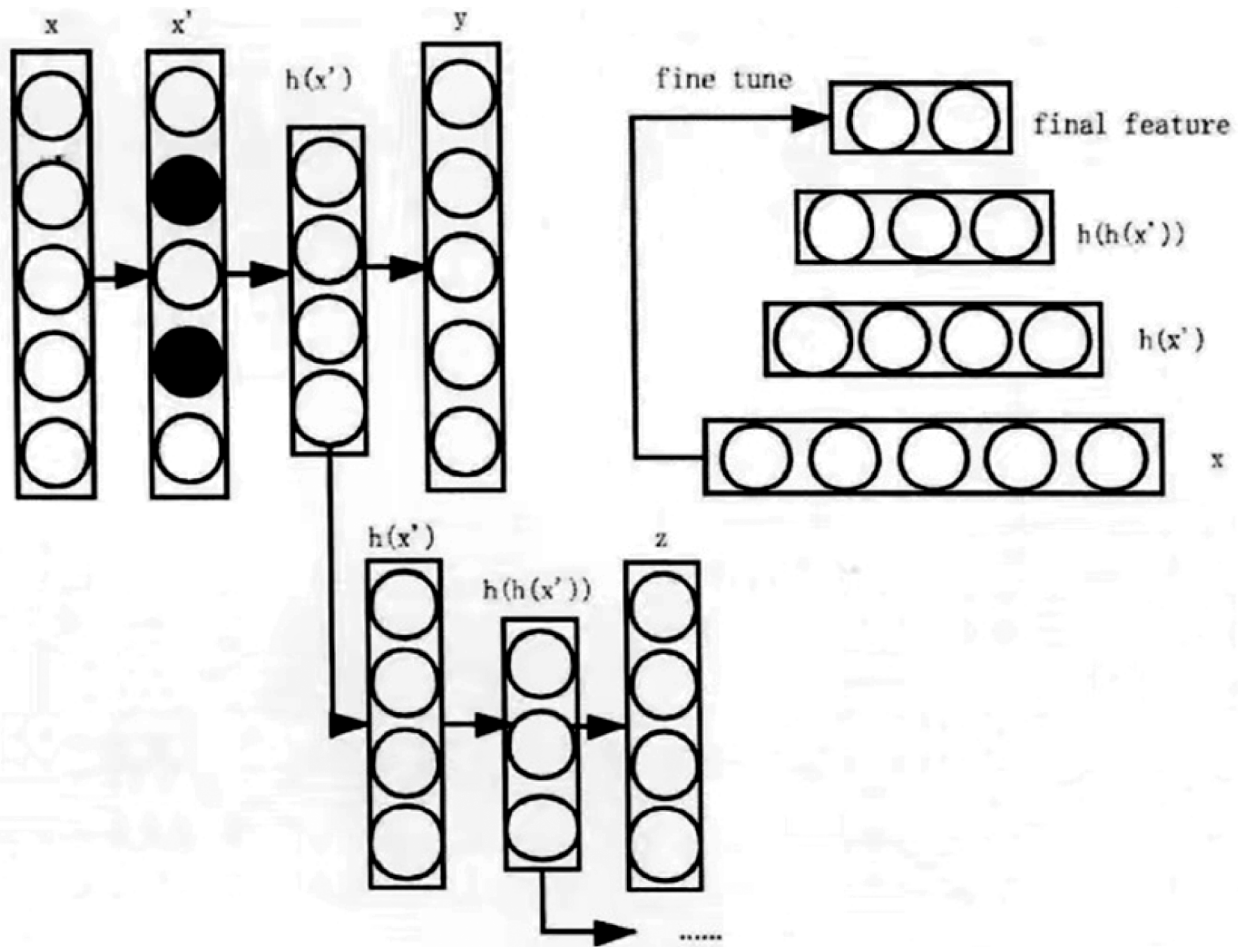


Рис. 3. Диаграмма процесса обучения самокодера глубокого шумоподавления

рейтинга пользователей  $r^{(u)} = (R_{u1}, \dots, R_{un}) \in R^n$ , набор тестовых данных  $T$ . Выход алгоритма: Предсказанная реконструированная матрица рейтинга; средняя абсолютная ошибка MAE и средняя квадратичная ошибка RMSE алгоритма тестового набора. Алгоритм рекомендации с самокодером глубокого шумоподавления [13–15]:

- Выполнить одно предсказание на исходной матрице с помощью улучшенного взвешенного алгоритма Slope one.
- Заполнить исходную матрицу соответствующими предсказанными значениями рейтинга.
- Произвести ввод матрицы оценок пользователей из новой матрицы оценок в самокодер с глубоким шумоподавлением.
- К данным во входном слое добавить маскирующий шум.
- Обучить самокодер глубокого шумоподавления в соответствии с послойным методом жадного обучения и определить соответствующие параметры модели путем непрерывной оптимизации функции потерь.
- На основе обученной модели восстанавливается матрица оценок на обучающем множестве  $T$  для получения прогнозных значений оценок.

- Прогнозируемые оценки сравниваются с исходными оценками и рассчитываются и выводятся средняя абсолютная ошибка MAE и среднеквадратичная ошибка RMSE.

#### Анализ численных данных

Для оценки качества работы модели был взят набор данных MovieLens, а для сравнения выбраны несколько других моделей, а именно: вероятностная матричная факторизация (PMF) [9], рекомендация на основе совместной фильтрации пользователя (U-KNN) [10], User-Based Restricted Boltzman Machines for Collaborative Filtering (U-RBM) [11], User-Based Autoencoder Recommendation Algorithm (U-AE) [12-14] и Noise Reduction Autoencoder Recommendation Algorithm (DAE), содержащий скрытый слой [15]. Т.е., необходимо было сравнить эффективность рекомендаций этих методов на одном и том же наборе данных.

Были выбраны два набора данных с различными уровнями разреженности данных: MovieLens-100K и MovieLens-1M соответственно.

Сравнение результатов предварительной обработки заполнения данных

Режим работы модели	MovieLens-100K	MovieLens-1M
Матрица оценок не заполнена заранее	0.675	0.811
Матрица оценок предварительно обрабатывается	0.731	0.855

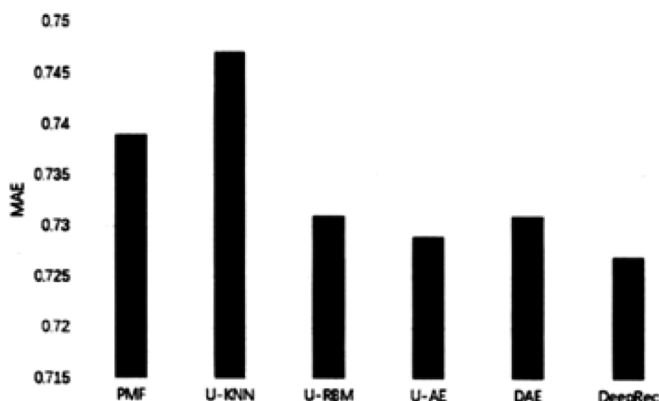


Рис. 4. Сравнение MAE DeepRec на наборе данных MovieLens-100K

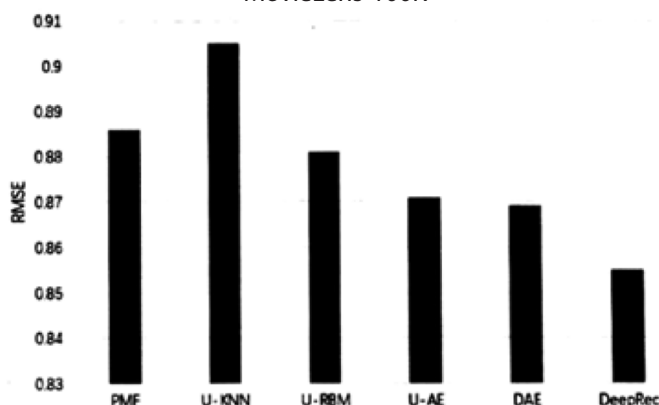


Рис. 5. Сравнение MAE DeepRec на наборе данных MovieLens-1M

Для экспериментов использовалась одна метрика, средняя абсолютная ошибка (MAE). В отличие от среднеквадратических ошибок, где используется квадрат разности, MAE является линейной оценкой, поэтому вес разностей одинаков независимо от диапазона. Это делает линейную шкалу более наглядной при сравнении точности моделей на разных масштабах данных.

После анализа результатов экспериментов можно сделать вывод, что для выбранного алгоритма средняя абсолютная ошибка является наименьшей. Т.е., алгоритм вполне приемлем для использования для решения задачи рекомендаций.

### Заключение

В данной работе авторы провели обзор алгоритма персонализированных рекомендаций на основе самокодера глубокого шумоподавления для фильмов, названный DeepRec. В настоящее время большинство моделей глубокого обучения предлагают выполнять анализ настроений, предполагая, что текст имеет статический характер. Однако по-прежнему сложно справиться с динамическим анализом настроений и отслеживанием, которое связано с динамическим характером потоковой передачи данных, как в социальных сетях. Давайте возьмем пример беседы в Twitter, в которой многие пользователи обсуждают выборы. Такого рода ситуация ставит перед анализом настроений следующие задачи. Во-первых, наиболее сложной проблемой является быстрое изменение набора данных на каждом временном шаге. Во-вторых, словари, используемые пользователями, могут меняться с течением времени. В-третьих, количество пользователей, участвующих в разговоре, и их предпочтения могут меняться с течением времени. Таким образом, проведение анализа настроений в такого рода ситуациях все еще остается открытой исследовательской проблемой. Насколько нам известно, для решения этой темы не было предложено никакой модели глубокого обучения. Таким образом, благодаря возможностям моделей глубокого обучения, особенно механизма самовнимания и рекуррентной нейронной сети моделирования последовательного ввода, мы предполагаем, что моделирование социальных сетей с помощью механизма самовнимания или рекуррентных нейронных сетей путем объединения ссылок на настройки и контента поможет справиться с этой ситуацией. Для проверки качества работы алгоритма были проведены эксперименты на наборе данных MovieLens и проведено сравнение с другими алгоритмами. Сравнение экспериментальных результатов алгоритма с несколькими другими алгоритмами показало, что алгоритм DeepRec имеет немного лучшее качество рекомендаций. В дальнейшем авторы планируют развить эту тему и для других предметных областей.

### ЛИТЕРАТУРА

1. Афанасьев Г.И. Афанасьев А.Г., Бурмистрова М.В. Тэт В.Я.С. Исследование методов машинного обучения для прогнозирования эффективных бизнес-решений в системах электронной коммерции // E-Scio. 2022. № 11 (74). С. 59–72.
2. Афанасьев Г.И. Афанасьев А.Г., Зо Х.А. Анализ возможностей применения компьютерного зрения в рекомендательных интернет-системах// E-Scio. 2022. № 11 (74). С. 73–82.
3. Афанасьев Г.И., Гой А.В., Ковалева Н.А. Анализ и сравнение методов Process Mining // Аспирант и соискатель. 2020. № 1 (115). С. 195–199.
4. Ч.Чжан, Афанасьев Г.И. Основные технологии и перспективы эволюции персонализированных рекомендательных систем//E-SCIO. 2022. №4(67). С.309–320.

5. Ч. Чжан, Афанасьев Г.И. Применение моделей глубокого обучения в области рекомендательных систем, основанных на содержании // Искусственный интеллект в автоматизированных системах управления и обработки данных. Сборник статей Всероссийской научной конференции. В 2-х томах. Москва: МГТУ им. Н.Э. Баумана. 2022. Т.1. С.278–284.
6. Steck H., Baltrunas L., Elahi E., Liang D., Raimond Y., & Basilico // Deep learning for recommender systems: A Netflix case study. // AI Magazine.2021. 42(3). P. 7–18.
7. Van den Oord A, Dieleman S, Schrauwen B. Deep content-based music recommendation[C] //Advances in neural information processing systems.2013. P. 2643–2651.
8. Nguyen H T H, Wistuba M, Grabocka J, et al. Personalized deep learning for tag recommendation //Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer. Cham. 2017. P.186–197.
9. Mnih A., Salakhutdinov R.R. Probabilistic matrix factorization//Advances in neural information processing systems. 2008. P.1257–1264.
10. Moosavi-Dezfooli S.M., Fawzi A., Frossard P. Deepfool: A Simple and Accurate Method to Fool Deep Neural Networks // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. P. 2574–2582. DOI 10.1109/CVPR.2016.282.
11. Herlocker J L, Konstan J A, Borchers A, et al. An algorithmic framework for performing collaborative filtering///Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval.1999. P.230–237.
12. Narodytka N., Kasiviswanathan S.P. Simple Black-Box Adversarial Perturbations for Deep Networks // ArXiv preprint. 2016. DOI 10.48550/ARXIV.1612.06299.
13. Komkov S., Petiushko A. Advhat: Real-world adversarial attack on ArcFace Face ID system // 25th International Conference on Pattern Recognition (ICPR). 2021. P. 819–826. DOI 10.1109/ICPR48806.2021.9412236.
14. Sedhain S, Menon A K, Sanner S, et al. Autorec: Autoencoders meet collaborative filtering[C]//Proceedings of the 24th international conference on World Wide Web. 2015. P.111–112.
15. Wu Y, DuBois C, Zheng A X, et al. Collaborative denoising auto-encoders for top-n recommender systems//Proceedings of the Ninth ACM International Conference on Web Search and Data Mining.2016. P. 153–162.

---

© Ян Тяньци (2936703080@qq.com); Афанасьев Геннадий Иванович (gaipcs@bmstu.ru); Калистратов Алексей Павлович (akalistratov@gmail.com)

Афанасьев Арсений Геннадьевич (afanasievag@bmstu.ru)

Журнал «Современная наука: актуальные проблемы теории и практики»