Автоматизация и моделирование в проектировании и управлении. 2023. № 2 (20). С. 69-76.
ISSN 2658-3488 print, ISSN 2658-6436 online
Automation and modeling in design and management. 2023. № 2 (20). P. 69-76.

Управление в организационных системах

Научная статья
Статья в открытом доступе
УДК 004.658.2
DOI 10.30987/2658-6436-2023-2-69-76

ФОРМИРОВАНИЕ РЕКОМЕНДАЦИЙ ПО КОНТЕНТУ ДЛЯ ПОЛЬЗОВАТЕЛЕЙ ОБРАЗОВАТЕЛЬНЫХ ПОРТАЛОВ НА ОСНОВЕ МАШИННОГО ОБУЧЕНИЯ

Андрей Владимирович Аверченков[1] Татьяна Михайловна Геращенкова[2], Дмитрий Иванович Гончаров[3]
[1, 2, 3] Брянский государственный технический университет, г. Брянск, Россия
[1] mahar@mail.ru, http://orcid.org/0000-0000-0000-0000
[2] gerash-tatyana@yandex.ru, http://orcid.org/0000-0000-0000-0000
[3] jeriho32@yandex.ru, http://orcid.org/0000-0000-0000-0000

Аннотация. *В работе предлагается создание научного подхода к формированию рекомендаций по контенту в разрабатываемом образовательном портале для учащихся, а также специалистов, которые желают повысить свой уровень квалификации, либо людей, которые занимаются самообразованием. Актуальность исследования заключается в том, что в настоящее время индивидуализация процесса обучения является новым актуальным трендом развития образовательной деятельности. Она позволяет тонко выстраивать систему и ее содержание под требования и предпочтения каждого пользователя. В отличие от универсального подхода, на котором построены классические образовательные порталы, которые предоставляют курсы обучения, ориентированные на широкие массы и построенные по шаблону, разрабатываемый портал на принципе индивидуализации обучения, ориентирован на обучающегося, и такая система может быть дополнительно им настроена. Для построения рекомендаций необходимо проанализировать данные о пользователе, собираемые в процессе его работы в системе, а также данные о размещенных в системе учебных материалах, представленных в виде метаданных. В статье описывается процесс анализа данных о пользователях и учебных материалах для формирования рекомендаций образовательного контента для обучающегося. Сравнивается два подхода к поиску сходства между элементами проектируемого образовательного портала для построения рекомендаций по контенту и на примерах показывается их применимость в системах онлайн-обучения.*

Ключевые слова: рекомендательные системы, системы электронного обучения, графовые базы данных
Для цитирования: Аверченков А.В., Геращенкова Т.М., Гончаров Д.И. Формирование рекомендаций по контенту для пользователей образовательных порталов на основе машинного обучения // Автоматизация и моделирование в проектировании и управлении. 2023. №2 (20). С. 69-76. doi: 10.30987/2658-6436-2023-2-69-76.

Original article
Open Access Article

FORMING CONTENT RECOMMENDATIONS FOR USERS OF EDUCATIONAL PORTALS BASED ON MACHINE LEARNING

Andrey V. Averchenkov[1], Tatyana M. Gerashchenkova[2], Dmitry I. Goncharov [3]
[1,2,3] Bryansk State Technical University, Bryansk, Russia.
[1] mahar@mail.ru, http://orcid.org/0000-0000-0000-0000
[2] gerash-tatyana@yandex.ru, http://orcid.org/0000-0000-0000-0000
[3] jeriho32@yandex.ru, http://orcid.org/0000-0000-0000-0000

Abstract. *The paper proposes creating a scientific approach to forming content recommendations in the educational portal being developed for students, as well as for professionals who wish to improve their skills or people who are engaged in self-education. The relevance of the study lies in the fact that at present the learning process individualization is a new current trend in developing educational activities. It allows fine-tuning the system and its content to each user's requirements and preferences. Unlike the universal approach on which classical educational portals are based, which provide training courses focused on the general public and built according to a template, the portal being developed on the learning individualization principle is focused on the student, and such a system can be additionally customized by him. To give recommendations, it is necessary to analyse user data collected in the course of his work in the system, as well as data on educational materials placed in the system, presented in the metadata form. The article describes the process of analysing data about users and educational materials to form recommendations about the educational content for the students. Two approaches to searching for similarities between the elements of the projected educational portal are compared to give content recommendations and examples show their applicability in e-learning systems.*

Keywords: recommender systems, e-learning systems, graph databases.

For citation: Averchenkov A.V., Gerashchenkova T.M., Goncharov D.I. Forming content recommendations for users of educational portals based on machine learning. Automation and modeling in design and management, 2023, no. 2 (20). pp. 69-76. doi: 10.30987/2658-6436-2023-2-69-76.

Введение

Системы рекомендаций принадлежат к классу систем фильтрации информации, которые предназначены для прогнозирования предпочтений пользователей. Данное понятие было введено учеными П. Ресником и Х.Р. Вариантом [4]. Позднее появился механизм «совместной фильтрации», который был разработан в начале 1990-х гг. Д. Голдбергом с соавторами [5]. Х. Су или Т. Хошгорфтар исследовали построение контент-фильтрации с использованием байесовских моделей – модели разложения по сингулярным значениям [6].

Построенные модели формирования рекомендаций, которые основаны на одном из вышеупомянутом подходе, находят довольно частое применение в разработанных системах в настоящее время. Применение технологий машинного обучения для анализа данных является новым витком в развитии моделей построения рекомендаций. Они позволяют обработать данные, которые имеют большой объем и довольно сложную структуру взаимосвязей.

Проектируемый в работе образовательный портал состоит из большого набора как самих данных, таких как учебные материалы и пользователи системы, так и связей между ними, такие как история посещения различных курсов, оценки учебных материалов, взаимосвязи учебных материалов между собой для последовательного их изучения. В связи с большими объемами образовательных ресурсов, объем которых растет в экспоненциальной зависимости, их обработка и формирование рекомендаций классическими подходами, такими как совместная фильтрация или контент-фильтрация является довольно серьезной проблемой. Также анализируемые данные о пользователе разрабатываемого портала имеют сложную структуру и имеют сложные взаимосвязи, которые будут выступать в качестве исходного материала для построения рекомендаций пользователю. Анализ этих данных с использованием классических алгоритмов оказывает большое влияние на производительность системы и точность полученных результатов. Для эффективного решения такого рода задач прибегают к технологиям машинного обучения с использованием алгоритмов для разбиения большого множества элементов на сообщества, определения их четких границ, а затем формирования рекомендаций для элементов данного сообщества на основе применения математических моделей расчета их схожести.

Поиск сообществ

Первоначальным этапом анализа больших массивов данных в разрабатываемом портале является разбиение элементов системы на группы, имеющие схожие признаки, т.е. сообщества. В качестве сообществ в разрабатываемом портале обучения выступают пользователи, которые обладают схожей историей о пройденных курсах обучения, информацией об интересах в личном профиле, а также иными данными, на основании которых пользователей можно отнести к одной группе (сообществу). По аналогичному принципу происходит формирование групп из учебных материалов. Разбиение учебных материалов на группы осуще-

ствляется на базе обнаружения сходства в метаданных, которые являются источником первичной информации для процесса машинного обучения.

Обнаружение групп элементов системы со схожими признаками и связями между собой, называемых сообщества, может использоваться в сочетании с другими алгоритмами. Например, сначала используется алгоритм PageRank или Centrality для поиска интересующего объекта. Затем находится сообщество, к которому он принадлежит. Можно также поменять порядок. Сначала найти сообщества, затем подсчитать, сколько интересующих объектов находится в этом сообществе.

В науке о сетях и связях сообщество представляет собой набор вершин, которые имеют более высокую плотность соединений внутри группы, чем за ее пределами [2, С. 54]. В частности, сообщество анализируется на плотность ребер, которая представляет собой количество ребер между членами, деленное на количество вершин. Значительное изменение плотности ребер обеспечивает естественную границу между теми, кто в группе, и теми, кто не состоит в ней. Определяют несколько различных классов сообществ в зависимости от того, насколько плотно они связаны между собой. Более высокая плотность ребер подразумевает более устойчивое сообщество.

Следующие три типа сообществ охватывают диапазон от самой слабой плотности до самой сильной плотности и все, что находится между ними.

Connected component – к членам сообщества данного типа относятся такие компоненты, которые имеют хотя бы одну прямую связь с одним или несколькими членами сообщества.

K-Core – к таким сообществам относятся компоненты, имеющие прямую связь с k или более членами сообщества, где k – целое положительное число.

Clique – членами таких сообществ являются компоненты, имеющие прямую связь с каждым другим членом сообщества.

На рис. 1 показаны примеры этих трех сообществ. Элементы, входящие в сообщество выделены отличным от белого оттенком. В качестве примера схематично изображены три сообщества с различными связями. Элементы, не принадлежащие сообществу по определению типа, выделены белым цветом. Таким образом, в нашем примере к первому типу сообщества Connected component относятся все элементы, так как в каждом из приведенных сообществ его элементы имеют хотя бы одно или несколько связей с другими членами. Ко второму типу сообщества K-Core со степенью $k = 2$ не относятся элементы 1, 2, 7 и 12, так как они не имеют 2 связей внутри своих сообществ.

Модулярность сообществ

В различных сценариях при анализе сообществ не имеется в виду конкретно значение k при характеристике связей внутри него. Для того, чтобы система в процессе обучения самостоятельно определила плотность сообществ вводится характеристика сообщества, которая называется модулярность, которая рассматривает относительную плотность, сравнивая плотность внутри сообществ с количеством связей между сообществами. Стоит понимать, что модулярность является системой оценок, а не алгоритмом.

Лувенский алгоритм является одним из самых быстрых алгоритмом поиска границ сообществ с очень высокой модулярностью. Он работает иерархически, формируя небольшие локальные сообщества, затем заменяя каждое сообщество одной мета-вершиной и повторяя формирование локальных сообществ мета-вершин.

Поиск сходства

Поиск сходства является необходимым условием для решения трех задач машинного обучения – кластеризация, классификация и предсказание связей. Посредством кластеризации происходит объединение похожих вещей в одну группу. Стоит отличать сообщества от кластера. Сообщества основаны на плотности связей; кластеры основаны на сходстве сущностей. Классификация является процессом принятия решения о том, к какой из нескольких заранее определенных категорий следует отнести объект [2, С. 81]. Распространенный метаподход заключается в том, что предмет следует отнести к той же категории, что и аналогич-

ные предметы, которые имеют известную категорию. Предварительное предсказание связей подразумевает расчетное предположение о том, что связь, которая в настоящее время не существует в наборе данных, с большой долей вероятности будет существовать в будущем. Для того, чтобы определять, какие материалы будут релевантными для пользователей, необходимо эти связи просчитывать.

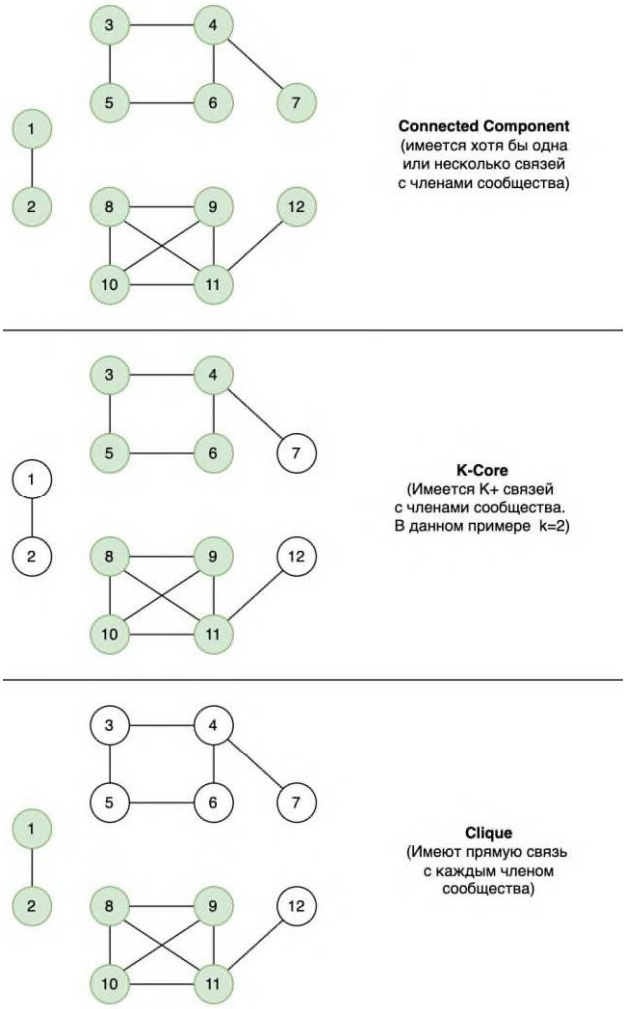

Рис. 1. Графическое изображение трех примеров сообщества, с самыми слабыми связями (сверху) и самыми сильными связями (снизу)
Fig. 1. Graphic representation of three examples of a community, with the weakest ties (top) and the strongest ties (bottom)

Сходство с соседями

Редко можно найти две сущности, которые имеют совершенно одинаковые окрестности. Двумя наиболее распространенными мерами для ранжирования сходства соседей являются сходство по Жаккарду и косинусное сходство.

Сходство Жаккарда. Сходство Жаккарда измеряет относительное перекрытие между двумя генеральными множествами [3, С. 152]. Предположим, что для образовательного портала необходимо провести аналитическое сравнение пользователей на основе того, какие курсы они завершили. В этом случае подойдет метод сходства Жаккарда: два множества – это наборы курсов, которые были пройдены каждым из двух сравниваемых пользователей. Чтобы сформулировать сходство по Жаккарду в общей терминологии, предположим, что два набора – это $N(a)$, окрестность вершины a, и $N(b)$, окрестность вершины b. Расчет сходства соседних элементов по Жаккарду:

$$jaccard(a,b) = \frac{|N(a) \cap N(b)|}{|N(a) \cup N(b)|} \quad (1)$$

Максимально возможная оценка равна 1, что произойдет при условии, если a и b имеют абсолютно одинаковых соседних элементов. Минимальная оценка – 0, если у них нет общих соседних элементов.

Рассмотрим следующий пример. Три пользователя, A, B и C, окончили следующие курсы, как показано в табл. 1.

Таблица 1

Набор данных для расчета сходства по Жаккарду

Table 1

A data set for calculating the similarity of Jacquard

Набор образовательных курсов / Отметка о прохождении курса	Пользователь A	Пользователь B	Пользователь C
Курс_1	✓	✓	
Курс_2	✓	✓	✓
Курс_3	✓		✓
Курс_4	✓	✓	
Курс_5	✓		✓
Курс_6		✓	
Курс_7		✓	
Курс_8			✓
Курс_9		✓	
Курс_10	✓		✓

Используя данные таблицы, вычислим сходство по Жаккарду для каждой пары пользователей.

– пользователь A и пользователь B имеют три общих пройденных курса (Курс_1, Курс_2, Курс_4). В совокупности они завершили 9 курсов. $jaccard(A, B) = 3/9 = 0{,}33$.

– у пользователя B и пользователя C только один общий пройденный курс (Курс_2). В совокупности они прошли десять курсов. $jaccard(B, C) = 1/10 = 0{,}10$.

– пользователь A и пользователь C имеют четыре общих пройденных курса (Курс_2, Курс_3, Курс_5, Курс_10). В совокупности они прошли семь курсов. $jaccard(A, C) = 4/7 = 0{,}57$.

Из этих трех пользователей пользователь A и пользователь C наиболее похожи. Поэтому система рекомендаций может предложить Пользователю C изучить некоторые курсы, которые прошел пользователь A, например, Курс_1 или Курс_4.

Косинусное сходство. Косинусное сходство измеряет выравнивание двух последовательностей числовых характеристик. Название происходит от геометрической интерпретации, в которой числовая последовательность – это координаты объекта в пространстве [3, С. 152]. Точки данных на сетке (другой тип графика) на рис. 2 иллюстрируют эту интерпретацию.

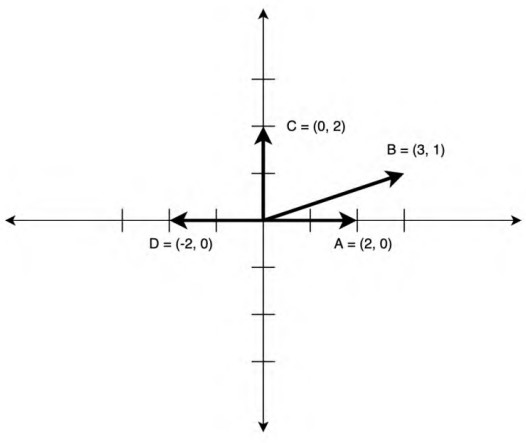

Рис. 2. Геометрическая интерпретация векторов числовых данных

Fig. 2. Geometric interpretation of numerical data vectors

Точка *A* представляет объект, вектор признаков которого равен (2, 0). Вектор признаков точки *B* равен (3, 1). Теперь понятно, почему мы называем список значений свойств «вектором». Векторы для *A* и *B* несколько выровнены. Косинус угла между ними – это их оценка сходства. Если два вектора направлены в одну сторону, угол между ними равен 0; косинус угла равен 1. cos(*A*,*C*) равен 0, потому что *A* и *C* перпендикулярны; у векторов (2, 0) и (0, 2) нет ничего общего. cos(*A*,*D*) равен -1, потому что *A* и *D* направлены в противоположные стороны. Таким образом, cos(*x*, *y*) = 1 для двух совершенно одинаковых объектов, 0 – для двух совершенно несвязанных объектов и -1 – для двух совершенно некоррелированных объектов.

Предположим, имеются оценки по нескольким категориям или атрибутам для набора сущностей. Необходимо объединить сущности в кластеры, поэтому нужна общая оценка сходства между сущностями. Продолжим пример с образовательным порталом. На этот раз каждый пользователь оценил курсы по шкале от 1 до 10, поэтому в качестве исходных данных имеются числовые значения (табл. 2).

Таблица 2

Набор данных для расчета косинусного сходства

Table 2

A data set for calculating cosine similarity

Набор образовательных курсов / Оценка курса	Пользователь *A*	Пользователь *B*	Пользователь *C*
Курс_1	8		
Курс_2	10	6	8
Курс_3	8		8
Курс_4	9	7	
Курс_5			4
Курс_6		7	
Курс_7		10	
Курс_8			10
Курс_9		9	
Курс_10	7		10

Ниже приведены шаги по использованию этой таблицы для расчета косинусного сходства между пользователями образовательного портала.

1. Перечислим всех возможных соседей и определим стандартный порядок для списка, чтобы можно было сформировать векторы. Будем использовать порядок сверху вниз, от Курса_1 до Курса_10.

2. Если граф имеет *D* возможных соседей, это дает каждой вершине вектор длины *D*. Для табл. 2 *D* = 10. Каждый элемент вектора – это либо вес ребра, если эта вершина является соседом, либо нулевая оценка, если она не является соседом.

3. Определение правильной нулевой оценки важно для того, чтобы оценки сходства означали именно то, что хотелось, чтобы они означали. Если 0 означает, что кому-то абсолютно не понравился курс, то будет неправильно ставить 0, если кто-то не изучал данный курс. Лучший подход – это нормализация оценок. Нормализовать можно либо по субъекту (пользователь), либо по соседу/характеристике (курс), либо по обоим параметрам.

Просто используем 1 в качестве оценки по умолчанию. Тогда вектор пользователя *A* будет *Wt*(*A*) = [8, 10, 8, 9, 1, 1, 1, 1, 1, 7]. Затем применим формулу в уравнении 2 для косинусного сходства: Идея заключается в том, чтобы заменить пустые ячейки оценкой по умолчанию. Можно установить значение по умолчанию как среднее значение оценки, а можно установить его немного ниже, так как отсутствие оценки о курсе – это довольно сильный минус в части популярности курса. Для простоты не будем нормализовывать оценки, будем просто использовать 1 в качестве рейтинга по умолчанию. Тогда вектор пользователя *A* будет *Wt*(*A*) = [8, 10, 5, 8, 9, 1, 1, 1, 1, 7], а пользователя *B* *Wt*(*B*) = [1, 6, 1, 7, 1, 7, 10, 1, 9, 1].

4. Затем применим формулу для расчета косинусного сходства соседей:

$$cosine(a,b) = \frac{Wt(a)Wt(b)}{\|Wt(a)\|\,\|Wt(b)\|} = \frac{\sum_{i=1}^{D} Wt(a)_i Wt(b)_i}{\sqrt{\sum_{i=1}^{D} Wt(a)_i^2}\sqrt{\sum_{i=1}^{D} Wt(b)_i^2}} \quad (2)$$

где $Wt(a)$ и $Wt(b)$ – это векторы веса соседних связей для a и b, соответственно.

Числитель проходит элемент за элементом по векторам, умножая вес от a на вес от b, а затем складывая их. Чем больше совпадают веса, тем большую сумму мы получаем. Знаменатель – это масштабный коэффициент, евклидова длина вектора $Wt(a)$, умноженная на длину вектора $Wt(b)$.

$$cosine(a,b) = \frac{(8\cdot 1)+(10\cdot 6)+(5\cdot 1)+(8\cdot 7)+(9\cdot 1)+(1\cdot 7)+(1\cdot 10)+(1\cdot 1)+(1\cdot 9)+(7\cdot 1)}{\sqrt{387}\cdot\sqrt{320}} \approx$$

$$\approx \frac{173}{351{,}9} \approx 0{,}491$$

Полученный показатель не может свидетельствовать об абсолютном сходстве пользователя A и пользователя B, но он означает наличие небольшого сходства, которое больше, чем у рассматриваемой случайной пары.

Заключение

Таким образом, в данной статье были рассмотрены способы формирования рекомендаций образовательного контента на основе двух математических моделей расчета схожести пользователей системы по их оценкам образовательных курсов с применением меры косинусного сходства, а также по статистике прохождения курсов с использованием меры сходства по Жаккарду. Полученные модели позволяют формировать рекомендаций образовательного контента в образовательных порталах на основе анализа оценок курсов других пользователей и статистики завершения обучения по тем или иным программам и использоваться в графовых базах данных в процессе машинного обучения. Полученные результаты расчета сходства между пользователями разрабатываемой системы, которые относятся к членам одной группы, позволяют сделать вывод о том, что предложенный подход применим к разрабатываемому образовательному порталу. Он также позволяет формировать рекомендации по контенту для пользователей разрабатываемого образовательного портала в условиях обработки большого объема данных.

Список источников:

1. Коннолли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. – М.: Вильямс И.Д., 2017. – 1440 с.
2. Anant Kumar. Graph Database Modeling with neo4j, Independently published, 2020, 124 p.
3. Victor Lee, Phuc Kien Nguyen, Xinyu Chang. Graph-Powered Analytics and Machine Learning with TigerGraph, Independently published, 2021, 223 p.
4. Resnick P., Varian, H.R. Recommender systems. Communications of the ACM, 40(3), p. 56-59.
5. Goldberg D., Nichols D., Oki B., Terry D. Using collaborative filtering to weave an information tapestry. Communications of the ACM, 1992, 35(12), p. 61-70.
6. Su X., Khoshgoftaar T.A survey of collaborative filtering techniques. Advances in Artificial Intelligence, 2009, p. 1-19.

References:

1. Connolly T. Databases. Design, Implementation and Support. Theory and Practice. Moscow: Williams I.D.; 2017.
2. Kumar A. Graph Database Modelling With Neo4j. Independently Published; 2020.
3. Lee V., Nguyen Ph. K., Chang X. Graph-Powered Analytics and Machine Learning With TigerGraph. Independently Published; 2021.
4. Resnick P., Varian H.R. Recommender Systems. Communications of the ACM, 40(3):56-59.
5. Goldberg D., Nichols D., Oki B., Terry D. Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM. 1992; 35(12):61-70.
6. Su X., Khoshgoftaar T. A Survey of Collaborative Filtering Techniques. Advances in Artificial Intelligence. 2009;1-19.

Библиографический список:

1. Мартишин С.А., Симонов В.Л., Храпченко М.В. Проектирование и реализация баз данных в СУБД MySQL с использованием MySQL Workbench: Методы и средства проектирования информационных систем и технолог. – М.: Форум, 2017. – 62 с.
2. Мартишин С.А., Симонов В.Л., Храпченко М.В. Проектирование и реализация баз данных в СУБД MySQL с использованием MySQL Workbench:

Bibliographic list:

1. Martishin S.A., Simonov V.L., Khrapchenko M.V. Design and Implementation of Databases in MySQL DBMS Using MySQL Workbench: Methods and Tools for Designing Information Systems and Technologies. Moscow: Forum; 2017.
2. Martishin S.A., Simonov V.L., Khrapchenko M.V. Design and Implementation of Databases in MySQL DBMS Using MySQL Workbench: Methods and

Методы и средства проектирования информационных систем и технологий. – М.: Форум, 2018. – 61 с.

3. Стружкин Н.П., Годин В.В. Базы данных: проектирование: Учебник для академического бакалавриата. – Люберцы: Юрайт, 2016. – 477 с.

Tools for Designing Information Systems and Technologies. Moscow: Forum; 2018.

3. Struzhkin N.P., Godin V.V. Databases: Designing. Lyubertsy: Yurayt; 2016.

Информация об авторах:

Аверченков Андрей Владимирович - доцент, доктор технических наук, заведующий кафедрой «Компьютерные технологии и системы»

Геращенкова Татьяна Михайловна - доцент, доктор экономических наук, проректор по качеству и аккредитации

Гончаров Дмитрий Иванович - аспирант

Information about authors:

Averchenkov Andrey Vladimirovich – Associate Professor, Doctor of Technical Sciences, Head of the Department «Computer Technologies and Systems»

Gerashchenkova Tatyana Mikhailovna – Associate Professor, Doctor of Economic Sciences, Vice-Rector for Quality and Accreditation.

Goncharov Dmitry Ivanovich – Postgraduate student

Вклад авторов: все авторы сделали эквивалентный вклад в подготовку публикации.

Contribution of the authors: the authors contributed equally to this article.

Авторы заявляют об отсутствии конфликта интересов.
The authors declare no conflicts of interests.

Статья поступила в редакцию 25.03.2023; одобрена после рецензирования 12.04.2023; принята к публикации 19.04.2023.

The article was submitted 25.03.2023; approved after reviewing 12.04.2023; accepted for publication 19.04.2023.

Рецензент – Фраймович Д.Ю., доктор экономических наук, профессор, Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых.

Reviewer –Fraimovich D.Yu., Doctor of Economic Sciences, Professor, Vladimir State University named after Alexander and Nikolay Stoletovs.