УДК 004.75

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПОДХОДОВ К УПРАВЛЕНИЮ БАЗАМИ ДАННЫХ ДЛЯ ОРГАНИЗАЦИИ ХРАНИЛИЩА РЕПОЗИТОРИЯ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ

Ямашкин С.А., Скворцов М.А., Большакова М.В., Ямашкин А.А.

ФГБОУ «Национальный исследовательский Мордовский государственный университет им. Н.П. Огарёва», Саранск, e-mail: dep-general@adm.mrsu.ru

Значительную роль в решении задачи усиления связанности территории Российской Федерации играет внедрение эффективных цифровых инфраструктур пространственных данных регионов России, нацеленных на оперативную диагностику природно-социально-производственных систем и высокоточное прогнозирование развития стихийных процессов и явлений. Ядро систем данного класса представляют методы и алгоритмы машинного анализа пространственных данных, позволяющие решать целый спектр прикладных задач – обнаружение аномалий, классификация данных, обучение признакам, объединение данных. Области применения результатов анализа в народном хозяйстве при этом чрезвычайно широки – от повышения эффективности сельского хозяйства до оценки последствий стихийных процессов. Цель исследования заключается в проведении сравнительного анализа подходов к управлению базами данных для организации хранилища репозитория нейросетевых моделей. Основное направление исследования направлено на изучение существующих видов и форм хранения информации, их классификации и анализа существующих решений. Подробный анализ существующих видов баз данных позволит выявить плюсы и минусы существующих решений, а так же подобрать решение максимально подходящее для хранения весов нейронных сетей. В результате данного исследования были проанализированы различные виды баз данных. Рассмотрены существующие на рынке программного обеспечения продукты и выделено три программных продукта подходящих для хранения весов нейронных сетей.

Ключевые слова: база данных, обмен информацией, хранение информации, хранение пространственных данных, инфраструктура пространственных данных, PostgreSQL, InfluxBD, Neo4j

COMPARATIVE ANALYSIS OF APPROACHES TO DATABASE MANAGEMENT FOR ORGANIZING A REPOSITORY OF NEURAL NETWORK MODELS

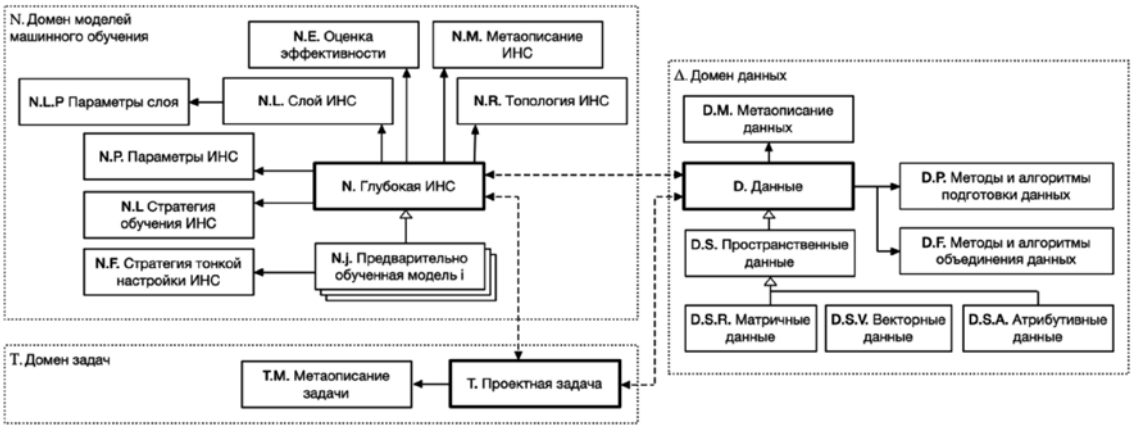
Yamashkin S.A., Skvortsov M.A., Bolshakova M.V., Yamashkin A.A.

National Research Ogarev Mordovia State University, Saransk, e-mail: dep-general@adm.mrsu.ru

A significant role in solving the problem of strengthening the connectivity of the territory of the Russian Federation is played by the introduction of effective digital infrastructures of spatial data of the regions of Russia, aimed at operational diagnostics of natural-social-production systems and high-precision forecasting of the development of natural processes and phenomena. The core of this class of systems is represented by methods and algorithms for machine analysis of spatial data, which allow solving a whole range of applied problems – anomaly detection, data classification, feature training, data fusion. The areas of application of the results of the analysis in the national economy are extremely wide – from increasing the efficiency of agriculture to assessing the consequences of natural processes. The purpose of the research is to conduct a comparative analysis of approaches to database management for organizing a repository of neural network models. The main direction of the research is aimed at studying the existing types and forms of information storage, their classification and analysis of existing solutions. A detailed analysis of existing types of databases will reveal the pros and cons of existing solutions, as well as choose a solution that is most suitable for storing the weights of neural networks. As a result of this study, various types of databases were analyzed. The products existing on the software market are considered and three software products are selected that are suitable for storing the weights of neural networks.

Keywords: database, information exchange, information storage, spatial data storage, spatial data infrastructure, PostgreSQL, InfluxBD, Neo4j

Значительную роль в решении задачи усиления связанности территории Российской Федерации играет внедрение эффективных цифровых инфраструктур пространственных данных (ИПД) регионов России, нацеленных на оперативную диагностику природно-социально-производственных систем (ПСПС) и высокоточное прогнозирование развития стихийных процессов и явлений. Ядро систем данного класса представляют методы и алгоритмы машинного анализа пространственных данных, позволяющие решать целый спектр прикладных задач – обнаружение аномалий, классификация данных, обучение признакам, объединение данных [1]. Предметом анализа могут выступать данные космической съемки, аэрофотосъемка, массивы информации об природных, социальных и экономических объектах, имеющих распределенную геопространственную организацию. Области применения результатов анализа в народном хозяйстве при этом чрезвычайно широки – от повышения эффективности сельского хозяйства до оценки последствий стихийных процессов.



Онтологическая модель репозитория для пространственного анализа и прогнозирования с применением глубоких нейронных сетей. Ключевые понятия

Задача проектирования и обучения эффективных глубоких нейросетевых моделей для анализа больших массивов пространственных данных встречает перед собой множество проблемных моментов, требующих поиска решений. Научная проблема накопления и систематизации моделей и алгоритмов машинного обучения с целью поддержки процесса принятия управленческих решений в области обеспечения условий устойчивого развития регионов России может получить решение благодаря разработке и внедрению репозитория глубоких нейросетевых моделей для анализа и прогнозирования развития пространственных процессов.

Цель исследования заключается в проведении сравнительного анализа подходов к управлению базами данных для организации хранилища репозитория нейросетевых моделей. Основное направление исследования – на изучение существующих видов и форм хранения информации, их классификации и анализа существующих решений. Подробный анализ существующих видов баз данных позволит выявить плюсы и минусы существующих решений, а также подобрать решение, максимально подходящее для хранения весов нейронных сетей.

Материалы и методы исследования

Решение задачи формирования архитектуры и программной реализации репозитория глубоких нейросетевых моделей должно опираться на онтологическую модель, определяющую формализованное описание топологий глубоких моделей, решаемых задач, множества анализируемых данных, алгоритмов обучения, а также отношений между этими сущностями. Онтологическая модель репозитория выдвигает требования к хранилищу данных и может быть декомпозирована на домены моделей глубокого машинного обучения, решаемых задач и данных и позволит дать комплексное определение формализуемой области знаний: каждая хранимая модель будет сопоставлена с набором конкретных задач и наборами данных (тензорных, растровых, векторных, атрибутивных) (рисунок).

Данная организация репозитория позволит дать формализованное определение исследуемых знаний, поможет сформировать базу для проектирования платформенного решения консолидации, подбора, эффективного использования и хранения нейросетевых моделей для решения проблемно-ориентированных задач.

Нейросетевые модели являются одной из перспективных технологий, текущий уровень развития которой поражает воображение и притягивает к себе все больше и больше внимания. Нейронные сети (НС) представляют собой совокупность моделей биологических нейронных сетей. Обучение нейросетевых моделей – это задача многомерной оптимизации, и для ее решения существует множество алгоритмов. Для обучения нейронных сетей необходимо собирать, сортировать и хранить датасет. Датасет – это обработанная и структурированная информация в табличном виде. После обучения нейронной сети мы получаем готовый набор весов, с помощью которых мы можем восстановить обученную нейронную сеть. Процесс обучения многогранен, и существует риск как недоучить нейронную сеть, так и переобучить ее. Поэтому принято сохранять веса во время обучения либо по таймеру эпох, либо по наилучшим результатам. Таким образом возникает большое количество данных, которое не-

обходимо хранить. Для автоматизации процесса обучения придется придумать грамотный процесс сохранения полученных данных [2].

П.А. Клеменков и С.Д. Кузнецов в своей статье «Большие данные: современные подходы к хранению и обработке» [3] анализировали возможные способы решения проблем с хранением и обработкой больших данных, а также рассмотрели три современных подхода к работе с большими данными. В статье «Big Data – большие данные в бизнесе» [4] автор рассмотрел различные возможности использования больших данных, проанализировал варианты использования сервисов для хранения больших данных. Е.П. Гордиенко и Н.С. Паненко В статье «Современные технологии обработки и анализа больших данных в научных исследованиях» [5] также рассматривали вопрос анализа хранения больших данных. Был выполнен обзор технологий больших данных и рассмотрены перспективы развития методов анализа больших данных в научных исследованиях.

Для систематизации процесса сохранения данных логичнее всего будет воспользоваться помощью базы данных (БД). Для начала введем определение БД – это некая структурированная форма представления информации, которая необходима для изменения, обработки и хранения взаимосвязанной информации, как правило, больших объемов. Семейства БД, называемых также моделями БД, представляют собой структуры и шаблоны, используемые для организации данных в системе управления базами данных (СУБД). Тип влияет на то, как будут представлены данные, какие операции сможет выполнять приложение, на функции СУБД для разработки и запуска. В данной статье мы рассмотрим современные типы баз данных, проведем сравнительный анализ и постараемся определить наиболее подходящие БД для хранения нейронных сетей.

За долгое время существования баз данных они претерпели довольно большое количество изменений и доработок. Рассмотрим основные типы баз данных, существующих на данный момент:

1) Простейшие базы данных – это простейший способ хранения данных – текстовые массивы. Данная методика применяется и в наши дни для работы с малыми объёмами информации. Для разделения полей используется специальный символьный разделитель: пробел или двоеточие в *nix-подобных системах, точка с запятой или запятая в csv-файлах датасетов [6]. Они являются самыми простыми базами данных. Имеют одну из простейших структур и слабо подходят для хранения больших массивов данных. Среди примеров таких баз данных можно выделить такие, как csv-файлы, файловые системы, DNS, LDAP, IDMS.

2) Реляционные базы данных. К реляционным базам данных относятся в первую очередь SQL базы данных. Это старейший тип используемых баз данных. Основа организации – это таблицы и связи, установленные между ними. Каждая строка в таблице представляет элемент данных в таблице или отдельную запись, который содержит значение для каждого из столбцов. В качестве примера можно привести следующие базы данных – MySQL, MariaDB, PostgreSQL, SQLite.

3) NoSQL базы данных. Это группа баз данных, предлагающих подходы, не совсем вяжущиеся с подходом SQL БД. Когда говорят о таких БД, подразумевают подход расширяющий или полностью отличный от уже существующей SQL структуры. Существуют 5 основных типов NoSQL баз данных: документные базы данных; колоночные базы данных; базы данных «ключ-значение»; графовые базы данных; базы данных временных рядов. Такие базы данных в основе своей имеют структуру map-объекта, имеющего два поля – ключ и значение. Отличие первого от второго типа заключается в формате данных, установленного в поле значение. Для документных баз поле значение содержит данные определенных жестко установленных типов, таких как JSON, BSON или XML. Примерами первых двух типов баз данных являются Redis, memcached, etcd, MongoDB, RethinkDB.

Дальнейшие типы баз данных представляют больший интерес для исследования. Они имеют более изощренную структуру. Данные в графовой базе данных представляют собой набор вершин и ребер между ними. Каждый узел в такой базе данных может иметь неограниченное количество связанных с ней узлов. Примерами таких баз данных могут являться Neo4j, JanusGraph, Dgraph. Колоночные базы данных относятся к типу NoSQL БД, но внешне схожи с реляционными БД. Как и реляционные, колоночные БД хранят данные, используя строки и столбцы, но с иной связью между элементами. Строка формируется из уникального строкового идентификатора, используемого для формирования поисковых запросов, за которым следуют наборы значений столбцов и имён. Следствием является то, что они удобны при работе с приложениями, требующими высокой производительности, данные и метаданные хранятся по одному идентификатору и га-

рантируют размещение данных из отдельно взятой строки в одном кластере, что упрощает сегментацию и масштабирование. Примерами таких баз данных являются Cassandra, HBase.

Последним и одним из самых интересных типов NoSQL баз данных являются базы временных рядов. Такие базы созданы для сбора и управления данными, меняющимися с течением времени. Для каждой записи в такой БД добавляется жесткая временная метка, которая характеризует состояние объекта в текущий момент времени. Для одной таблицы одновременно может поддерживаться несколько метрик.

Следствиями применения данного типа баз данных является ориентированность на запись, то, что они предназначены для постоянной обработки потока входных данных, производительность такой БД зависит от количества одновременно поддерживаемых метрик. Среди примеров можно выделить следующие – OpenTSDB, Prometheus, InfluxDB, TimescaleDB.

4) Комбинированные типы БД. Это интересный вид баз данных, созданный для того, чтобы извлечь максимальную выгоду из двух подходов к структурам БД – SQL и NoSQL. Существуют 2 типа таких БД: NewSQL базы данных; многомодельные базы данных. Они наследуют семантику и реляционную структуру, но построены с использованием масштабируемых конструкций. Главная цель – обеспечить более высокие гарантии согласованности, чем в NoSQL. Компромисс между согласованностью и большей масштабируемостью и доступностью, нежели реляционные БД, и является фундаментальной проблемой распределённых баз данных, описываемой в теореме CAP. Примерами таких БД выступают – Spanner, MemSQL, Calvin, VoltDB, CockroachDB, yugabyteDB, FaunaDB. Многомодельные базы данных – это гибридные базы данных, основанные на функциональных возможностях нескольких баз данных. Преимущества такого подхода в том, что для разных типов данных одна и та же система может использовать различные представления. Совместное размещение данных из нескольких типов БД в одной системе позволяет выполнять новые операции, которые в противном случае были бы затруднены или невозможны. Примерами таких БД являются – ArangoDB, OrientDB, Couchbase.

Результаты исследования и их обсуждение

Ознакомившись с основными типами баз данных, произведем подбор базы данных для хранения нейронных сетей. Нам нужно максимально оптимизировать процесс получения нейронной сети из базы данных, также нам бы хотелось поддерживать версионность хранимых весов, для записи всех весов, рассчитанных в процессе обучения.

Проведя небольшой анализ доступной документации, процента использования, потребности рынка было выделено несколько БД, достаточно подходящих под наши потребности: PostgreSQL, Neo4j, InfluxDB. Рассмотрим их более конкретно.

PostgreSQL – универсальный инструмент современного разработчика. Она имеет большое число поклонников, существуют версии почти для всех текущих операционных систем, имеет дружественный программный и графический интерфейс, надежна и имеет высокое быстродействие. Это одна из самых популярных баз данных, используемых в web-разработке.

Обращения к базе данных происходят посредством чистых SQL запросов. Однако кроме голого SQL существует огромное количество ORM систем, поддерживающих данную БД. Для справки – ORM (Object-Relational Mapping) – технология программирования, которая связывает базы данных с концепциями объектно-ориентированных языков программирования, создавая «виртуальную объектную базу данных» [7]. Тем самым во время разработки пользователю не придется спускаться в более низкоуровневый язык SQL, а писать команды на высокоуровневом языке разрабатываемой системы.

Несмотря на недостатки, данная БД является эталоном надежности и вполне подойдет для тривиальных задач, связанных с машинным обучением, а скорость ее работы и возможность гибкой настройки типов полей делают ее почти незаменимым и универсальным инструментом разработчика. Neo4j – графовая СУБД с открытым исходным кодом, реализована на Java компанией Neo Technology. Не уступает по производительности реляционным базам данных благодаря собственному формату хранения данных [8, 9].

Для работы с базой данных был разработан новый язык запросов – Chyper. Cypher – декларативный язык запросов в виде графа, позволяющий получить выразительный и эффективный запрос данных. Neo4j следует модели данных, называемой моделью графа собственных свойств. Здесь граф содержит узлы (сущности), и эти узлы связаны друг с другом. Узлы и отношения хранят данные в парах ключ-значение, известных как свойства. В Neo4j нет необхо-

димости следовать фиксированной схеме. Можно добавить или удалить свойства согласно требованию. Возможно масштабировать базу данных, увеличивая количество операций чтения/записи и объем, не влияя на скорость обработки запросов и целостность данных. Neo4j также обеспечивает поддержку репликации для обеспечения безопасности и надежности данных [10].

После рассмотрения данной БД сложно сказать о ее применимости к нейронным сетям. Так или иначе, она не решает проблемы хранения больших данных, но за счет графовой структуры мы можем напрямую создать структуру весов, тем самым снизив объем одного узла. Это обеспечит совершенно другой подход к структуре хранения весов нейронной сети, но значительно усложнит процесс внесения базы данных. Такой подход подойдет для внесения уже готовой и предобученной нейронной сети. Либо же придется разработать метод развертывания нейронной сети в базе данных после ее непосредственного локального обучения.

InfluxDB – это база данных, адаптированная для хранения статистических данных (параметры и их значения на определенный момент времени). Хорошо подходит для систем мониторинга, интернета вещей, различных метрик приложений. В базе данных InfluxDB хранятся точки. Точка имеет четыре компонента: измерение, набор тегов, набор полей и timestamp. Измерение позволяет связать точки, которые могут иметь разные наборы тегов или наборы полей. Набор тегов – это словарь пар ключ-значение для хранения метаданных с точкой. Набор полей представляет собой набор типизированных значений scalar-данных, записываемых точкой.

Каждая точка хранится ровно в одной базе данных в рамках ровно одной политики хранения. База данных – это контейнер для пользователей, политик хранения и точек. Политика хранения определяет, как долго InfluxDB хранит точки (продолжительность), сколько копий этих точек хранится в кластере (коэффициент репликации) и временной диапазон, охватываемый группами сегментов (продолжительность группы сегментов). Политика хранения позволяет пользователям легко (и эффективно для базы данных) удалять старые данные, которые больше не нужны. Это общая закономерность в приложениях временных рядов.

Когда база данных получает новые точки, она должна сделать эти точки долговечными, чтобы их можно было восстановить в случае сбоя базы данных или сервера и сделать точки доступными для запроса. Приложения временных рядов часто вытесняют данные из хранилища через определенный промежуток времени. Многие приложения мониторинга, например, будут хранить последние месяц или два данных в интернете для поддержки запросов мониторинга. Он должен быть эффективным для удаления данных из базы данных, если срок действия настроенного time-to-live истекает. Удаление точек из столбчатого хранилища обходится дорого, поэтому InfluxDB дополнительно организует свой столбчатый формат в ограниченные по времени фрагменты. Когда time-to-live истекает, ограниченный по времени файл может быть просто удален из файловой системы, а не требовать большого обновления сохраненных данных. За счет использования данной БД мы можем производить сохранение статистики появления всех весов, следить в реальном времени за процессом обучения нейронной сети и иметь доступ до любых обученных весов в любой момент времени.

Заключение

Внедрение в репозиторий моделей машинного обучения разрешит не просто построить банк глубоких ИНС, в области анализа пространственных данных различного типа разработанных для решения прикладных задач, но так же с помощью разработки системы рекомендаций и развертывания экспертного инструментария решить проблему подбора действенной модели, осуществляющего выбор наилучшего из алгоритмов. Каждую глубокую нейронную сеть необходимо опробовать на тестовых полигонах с целью поиска численных и субъективных оценок ее эффективности. Онтологическая модель репозитория определяет структуру хранилища данных и может быть декомпозирована на домены моделей глубокого машинного обучения, решаемых задач и данных, что позволит дать комплексное определение формализуемой области знаний.

Рассмотрение различных видов современных баз данных позволило выделить три наиболее подходящих для дальнейших разработок. Отметим, что каждая в отдельности парадигма к организации хранилища репозитория глубоких моделей машинного обучения не дает ответа на все вопросы, которые возникают при решении проблемы систематизации информации из доменов данных, моделей и проектных задач. Комплексный ответ на обозначенную проблему способны предоставить мультимодельные системы управления базми данных, представляющие собой гибридные хранилища, которые могут быть централизованы в цен-

тре обработки данных, или же представлены в масштабах облака, функционирование которых основано на суперпозиции возможностей, заложенных в СУБД разных классов.

Результатом грамотного использования мультимодельных систем управления данными репозитория глубоких нейросетевых моделей должно стать целенаправленное усиление качественных характеристик формируемого хранилища нейронных сетей, в том числе масштабирования и модульности, отказоустойчивости и надежности.

Работа выполнена при финансовой поддержке гранта Президента Российской Федерации (грант № МК-199.2021.1.6).

Список литературы

1. Ямашкин С.А., Ямашкин А.А., Занозин В.В. Формирование репозитория глубоких нейронных сетей в системе цифровой инфраструктуры пространственных данных // Потенциал интеллектуально одарённой молодежи – развитию науки и образования: материалы IX Международного научного форума молодых ученых, инноваторов, студентов и школьников. 2020. С. 370–375.

2. Федюшкин Н.А., Ямашкин С.А. Исследование репозиториев моделей нейронных сетей Научно-технический вестник Поволжья. 2021. № 3. С. 28–30.

3. Клеменков П.А., Кузнецов С.Д. Большие данные: современные подходы к хранению и обработке // Труды Института системного программирования РАН. 2012. Т. 23.

4. Сизов И.А. Big Data – большие данные в бизнесе // Экономика. Бизнес. Информатика. 2016. Т. 2. № 3. С. 8–23.

5. Гордиенко Е.П., Паненко Н.С. Современные технологии обработки и анализа больших данных в научных исследованиях // Актуальные проблемы железнодорожного транспорта. 2018. С. 44–48.

6. Савоськин И.В., Фирсов А.О. Исследование способов применения NoSQL и реляционных баз данных // E-Scio. 2019. № 6 (33).

7. Halpin T. Object-role modeling (ORM/NIAM). Handbook on architectures of information systems. Springer, Berlin, Heidelberg, 1998. P. 81–103.

8. Miller J.J. Graph database applications and concepts with Neo4j. Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA. 2013.

9. Naqvi S.N.Z., Yfantidou S., Zimányi E. Time series databases and influxdb. Studienarbeit, Université Libre de Bruxelles. 2017. P. 12.

10. Sahatqija K. et al. Comparison between relational and NOSQL databases. 2018 41st international convention on information and communication technology, electronics and microelectronics (MIPRO). IEEE, 2018. P. 216–221.