

Заметки о методах преподавания технологий обработки больших данных в прикладных областях

Мирошниченко И.Д., СПбГУ, Санкт-Петербург i.miroshnichenko@spbu.ru,
Султонов А.Ш., СПбГУ, Санкт-Петербург st138466@student.spbu.ru

Аннотация

Преподавание технологий обработки больших данных на математико-механическом факультете и факультете социологии, который представляет прикладную область с точки зрения специалистов разработки программного обеспечения, безусловно, отличается.

Статья посвящена особенностям подачи материала обучающимся старших курсов бакалавров и магистров факультета социологии или других направлений образовательных программ, цель которых — прикладное использование инструментов программного обеспечения, главным образом, изучение, исследование, применение специализированных пакетов, фреймворков и языков R и/или Python для решения реальной задачи моделирования, таким образом, чтобы компенсировать недостатки базовых системных знаний и понятий о структурах данных и параллельных процессах, происходящих при обработке больших данных и плохо формализованных подходах.

Введение

На протяжении почти двадцати лет значимость технологий, опирающихся на трансформацию Больших данных, всё больше увеличивается, эти технологии развиваются и усложняются, всё более широко проникают в самые различные области применения. В настоящее время методы обработки информации Больших данных, методы машинного обучения, нейронных сетей применяют при разработке проектов не только студенты естественно-научных факультетов, но и гуманитарных, решая задачи в прикладных областях цифровой экономики. Однако студентам гуманитарных факультетов или кафедр для получения полноценных профессиональных компетенций в области IT-технологий требуются более глубокие знания о структурах данных, принципах взаимодействия и особенностях функционирования потоков при распараллеливании в различных средах (с общей памятью и распределённых), способах передачи данных, платформах и интерфейсах, библиотеках, инструментах, видах облачных хранилищ и механизмах хранения или обработки

данных в них. Рассмотрение особенностей преподавания с этой точки зрения IT-дисциплин и является целью данной работы.

Различия в подходах к обучению

Понятно, что на гуманитарных факультетах преподаватели-специалисты, главным образом, обучают навыкам *применения* конкретных *методов* для решения конкретных задач проблемной области, а также изучают функции языков R и Python с точки зрения использования.

Это главное отличие направленности изучения методов обработки информации и структуры и функционирования языков программирования на естественно-научных факультетах (в частности на математико-механическом) и гуманитарных. На математико-механическом факультете как изучение основ программирования, так и методик, технологий, строится на детальном рассмотрении базовых структур, процессов, связей на всем процессе изучения различных дисциплин, на гуманитарных факультетах внимание обращается на возможности и набор средств (функций, пакетов), применяемых для конкретных целей в решении задач предметной области. Поэтому приходится обращать внимание на основы: на вид и представление данных в памяти, на особенности построения и функционирования конструкций, исключительные и критические состояния, на архитектуру аппаратных средств и влияние их конфигурации на ускорение вычислений или процесса взаимодействий, на необходимость, значимость и суть стандартов на всех уровнях от элементов архитектуры до построения инфраструктуры программного обеспечения.

Первый этап в освоении Big Data — изучение способов сбора, очистки, проверки на валидность больших данных и подготовки информации к дальнейшим преобразованиям — в основном, в изучении сложности не представляет, здесь, скорее, требуется исследование соответствующих пакетов, применяемых в конкретной прикладной области. Следующие этапы: способы и методы хранения и обработки — наоборот, включают в себя понимание принципов функционирования *распределенных систем хранения данных* и обработки запросов, а также основ механизмов работы с облачными платформами, которые предоставляют необходимую масштабируемость и гибкость для работы с большими данными. Кроме технических аспектов, важно не только применять, но и понимать суть методов анализа данных, включая *статистический анализ*, *машинное обучение* и методы визуализации данных, чтобы извлекать ценную информацию из больших объемов данных.

Как уже упоминалось, основные языки, используемые в работе с Big Data, — это Python и R. На разных платформах в интернет-пространстве орга-

низованы мощные библиотеки (в том числе большое количество авторских свободного доступа) для анализа данных и машинного обучения. Обычно в учебном процессе прикладной области используется достаточно большой набор известных пакетов (кроме базовых, предлагаемых средой моделирования), а в них рассматриваются конкретные команды (например, визуализации или анализа) с конкретными, наиболее часто используемыми, параметрами. В курсах, касающихся больших данных, приходится посвящать время тому, чтобы научить анализировать не только информацию, предоставленную автором в документации о методе, но и предварительно исследовать правильность его работы на разных типах данных, его сходимость, обусловленность и другие характеристики прежде, чем применить в собственном проекте.

Для того, чтобы разработать эффективный проект, используют такие инструменты, как *Apache Hadoop* и *Spark*, *NoSQL* базы данных, которые представляют собой основу для обработки, интеграции и анализа больших объемов данных, обеспечивая необходимую масштабируемость и гибкость. Также важным аспектом обучения является овладение навыками визуализации данных с помощью инструментов, таких как *Tableau* и *PowerBI*, что позволяет эффективно представлять и интерпретировать сложные наборы данных. Самостоятельная работа над проектом позволяет студентам применять изученные инструменты и методы в контексте реальных задач, развивает критическое мышление и аналитические навыки. Функция преподавателя в описываемых условиях – научить ориентироваться в обилии предоставляемой информации, критически отбирая наиболее подходящие инструменты к прикладной области, а также умению проводить сравнительный анализ их применимости к реальной задаче, профессиональному отбору параметров.

Необходимость фундаментальных знаний

Восполнение пробелов в базовых знаниях

Но для самостоятельной реализации поставленной задачи, оптимальной разработки структуры проекта, грамотного написания сценария нужны структурные основы базовых знаний. Восполнить эти пробелы — задача преподавателей математико-механического факультета, ведущих дисциплины, главным образом, на старших курсах. В частности, это касается, например, технологий, включающих понятия *распараллеливания* и/или *взаимодействия потоков* или детального объяснение понятий и особенностей моделирования и функционирования процессов, без которых вряд ли можно полноценно объяснить особенности структуры и функционирования *ETL* и *ELT* (основные механизмы обработки больших данных в процессе извлечения зна-

ний (*Data Mining*), машинного обучения, видов глубокого обучения.

Наконец, следует отметить, что ограничения в емкости устройств хранения информации сейчас уже не играют определяющей роли, значительно важнее становятся временные характеристики вычислителей, и это требует внедрения в практику многопроцессорных распределенных систем. Возникает необходимость изучения специальных методов обработки информации, учитывающих эти обстоятельства.

Корректировка учебного плана: основные направления

Отметим ниже то новое, что пришлось ввести в текущий процесс (не меняя содержательную часть РПД дисциплин) при подготовке материала, касающегося изучения среды программирования R и Python и работы с большими данными:

1. В преподавании дисциплин, связанных с языками программирования R и Python, популярных в обработке больших данных и предоставляемых на бесплатных платформах (например, в тематике «среда R и доступ к данным»), нужно обращать внимание на то, что эти языки имеют свойства как *функциональных языков, так и объектно-ориентированных*, и показать, в чём это проявляется. Кроме того, часто приходится объяснять свойства этих парадигм на примерах, так как, в общей массе, обучающиеся эти понятия плохо представляют. В частности, важно отметить, что, ввиду свойства функциональности, переменной, как таковой, в языке нет. Есть объект, который строится из одномерного вектора с единственной компонентой, и объект не изменяется. При конструировании нового объекта из старого объекта, у объекта меняется ссылка на новое значение.

Написание грамотных структурных сценариев – немаловажный пункт обучения программированию (и в прикладной области). Студенты не придают этому должного внимания, однако это важный момент, ускоряющий процесс написания, понимания и отладки сценария. Особенность парадигмы функционального программирования, так называемые, *ленивые вычисления*, неявно используются в реализациях сценариев на языках R и Python, но позволяют написать грамотные сценарии, реализующие сложные итеративные бесконечные методы. Таким образом, знание особенностей структур языка позволяет написать эффективный сценарий.

2. При изучении технологии работы с большими данными также оказывается полезным напомнить (или заново объяснить) структуру таких объ-

ектов как *хранилище*, *витрина*, *озеро*, по каким принципам строятся различные типы баз данных, каковы их свойства. Обычно обучающиеся приходят с навыками работы с реляционными базами данных, но с другими типами баз данных знакомы плохо. А для понимания построения, функционирования хранилищ, витрин, озер важно иметь представление:

- о многомерных базах данных, гибридных структурах,
 - базах данных NoSQL, их функциональных особенностях,
 - структурированных и неструктурированных,
 - какие инструменты работают с теми или иными типами данных.
3. Важная тема, на которую надо обратить внимание — это *открытые системы*. Хотя в учебных планах явно могут не присутствовать пункты, касающиеся базовой трехуровневой модели открытых систем, оказывается полезным напомнить или объяснить:
- что представляет собой протокол и интерфейс, а также семиуровневая модель передачи данных,
 - понятия масштабирования и балансировки в многопроцессорных и распределённых системах.

которые весьма важны для понимания и общего представления современной инфраструктуры программного обеспечения. Этих понятий практически не касаются или недостаточно подробно изучают в профильных дисциплинах гуманитарных факультетов.

4. Хотя в профильных дисциплинах изучаются (или ранее в предыдущих курсах, возможно, изучались более подробно) плохо формализуемые подходы к решению задач, а именно, нейронные системы, генетические алгоритмы и нечеткие подходы, тем не менее, весьма полезно детально *проработать методику решения задач* с подробным разбором примеров, в особенности, *генетические алгоритмы*, необходимые в исследовании Интернет-пространства, и *нечеткие системы*, нужные для исследования систем с плохо формализуемыми входными и/или выходными данными.

Таким образом, в современных реалиях в лекциях по дисциплинам, связанным с большими данными, для гуманитариев требуется подробно объяснять студентам необходимость более детально (хотя в плане и содержательной части дисциплин предполагается прикладная составляющая, например, «доступ к данным из сети Интернет»)

- изучать технологии *MapReduce* и *Data Mining* (модель распределённых вычислений, представленная компанией Google, используемая для параллельных вычислений над очень большими наборами данных в компьютерных кластерах).
- анализировать функциональную структуру пакета утилит и библиотек *Hadoop* (используемого для построения систем, обрабатывающих, хранящих и анализирующих большие массивы нереляционных данных, а именно, данные датчиков, интернет-трафика, объектов метаданных, файлов журналов, изображений и сообщений в соцсетях).
- использовать возможности фреймворка *Big Data Storm* (программной платформы, определяющей структуру программной системы, иначе, программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта и созданный для работы с информацией в режиме реального времени).
- облачную технологию *DataLake* (инструмент, который позволяет хранить любые данные: csv, xml, json, parquet, jpg, png, mov, mp3, pdf и другие). В него можно загружать таблицы, у которых нет чёткой структуры, то есть периодически меняется количество и названия колонок и строк. Все эти данные можно загружать в озеро без обработки, то есть практически мгновенно, включающую в себя программную платформу, источники и методы пополнения данных, кластеры узлов хранения и обработки информации, управления, инструментов обучения). *DataLake* при необходимости масштабируется до многих сотен узлов без прекращения работы кластера.

Педагогические выводы и значение самообразования

Методы, инструменты, технологии, отмеченных выше подходов и интеграция Big Data с передовыми технологиями такими, как искусственный интеллект и машинное обучение позволяют более эффективно анализировать и использовать большие объёмы данных, последние разработки и тенденции в этой быстро меняющейся цифровой инфраструктуре программного обеспечения.

Кроме того, непрерывное обучение и самообразование, чтение актуальной литературы и последних исследований в области обработки и анализа данных, играют важную роль в поддержании актуальности знаний и оставаться в курсе последних тенденций и инноваций в этой динамично развивающейся области. Реализация этого принципа в преподавании – в обязательной подготовке докладов на семинарах по современным технологиям.

Таким образом, материал дисциплин для факультета социологии и гуманитарных факультетов был серьезно переработан с учетом всех выше указанных пунктов в отличие от содержания материала для одноименных естественно-научных факультетов.

Заключение

В работе рассмотрены особенности преподавания дисциплин обработки и преобразования больших данных (актуальных сегодня на факультетах гуманитарных направлений для разработки проектов в различных прикладных областях) с точки зрения углубленности понимания внутренних процессов и инфраструктуры программного обеспечения, необходимых для повышения профессионализма специалиста. Для грамотной профессиональной подготовки студентов любой образовательной программы необходимо, *не ограничиваться прикладными наборами инструментов для разработки студенческих проектов*, необходимо иметь более глубокие знания в области как *структуры программного обеспечения, так и функциональности его отдельных инструментов и комплексного взаимодействия.*

Результаты могли бы быть полезны для построения соответствующих курсов гуманитарных направлений для магистров или бакалавров, что позволяло бы

- учитывать более профессионально особенности современного развития в области больших данных (и их роста),
- учитывать необходимость иметь представление о распределенных вычислениях, киберугрозах,
- разрабатывать проекты с учетом требований высокой доступности, адаптируемости к изменяющимся данным и их объёму, масштабируемости продуктов,
- иметь представление о подходах контейнеризации, микросервисных архитектур и облачных технологий.

Всё это могло бы помочь в выборе конкретного программного обеспечения для направлений разработки студенческих проектов.

С введением повсеместно в учебные планы дисциплин, связанных с обработкой больших данных, методов машинного обучения, алгоритмов исследования естественного языка с помощью искусственного интеллекта (не только как применение методов) становится еще более важно *грамотно и глубоко* изучать технические особенности рассматриваемых процессов.

Список литературы

- [1] Боровская Е.В., Давыдова Н.А. Основы искусственного интеллекта: учебное пособие. — М.: БИНОМ Лаборатория знаний, 2020. — 127 с.
- [2] IBM (2023). ELT vs. ETL: Similarities and Differences. <https://www.ibm.com/think/topics/elt-vs-etl>
- [3] AWS (2023). ETL and ELT design patterns for lake house architecture using Amazon Redshift: Part 1. AWS Big Data Blog. <https://aws.amazon.com/blogs/big-data/etl-and-elt-design-patterns-for-lake-house-architecture-using-amazon-redshift-part-1>
- [4] Освоение Big Data: стратегии и методы обучения для современных технологий // ЦифроОбраз. Образовательный портал про современное образование. — 20.12.2023. <https://iit-bsuir.by/tehnologii-budushhego/osvoenie-big-data-strategii-i-metody-obuchenija-dlja-sovremennyh-tehnologii/>
- [5] Степанов А.Г., Плотноков Г.А., Васильева В.С. Подходы к определению средств для построения методики обучения работе с большими данными // Информатика и образование. — 2021. — № 4 (323). — С. 54–61. <https://info.infojournal.ru/jour/article/view/681/538>
- [6] «А можно быстрее?»: разбираем методы ускорения обучения нейронных сетей // PVSM.RU. — 05.09.2024. <https://www.pvsm.ru/date/2024/09/05>
- [7] Зайнидинов Х.Н., Маллаев О.У., Зулунов Р.М., Нурмуродов Ж.Н. Методы *распараллеливания* процессов вычисления больших объемов данных с использованием технологий параллельного программирования // Автоматика и программная инженерия. — 2019. — № 4 (30). <http://www.jurnal.nips.ru/>
- [8] Мирошниченко И.Д., Загоревская Н.С. Задачи обучения особенностям параллельного программирования в среде MPI на базе кластерного компьютерного класса. — 2019. https://pureportal.spbu.ru/files/116708592/_2019_.pdf