оценки, так как этот признак не является актуальным для некоторого набора слов, например, местоимений, междометий и ряда других слов. Но это достаточно просто преодолеть, составив базу слов, которые необходимо заранее исключить из списка ключевых. Такую базу легко собрать, проанализировав большое количество разношерстных текстов, выделив *N*-е количество наиболее употребляемых слов. Чистый статистический метод может быть очень эффективен применительно к языкам со скудной морфологией, где каждое слово, вероятнее всего, не имеет огромного набора форм, например, в английском языке.

Литература

1. Мурашко, И. А. Оптимизация проектных решений : курс лекций для студентов специальностей 1-40 01 02 / И. А. Мурашко, Д. Е. Храбров. – Гомель : ГГТУ им. П. О. Сухого, 2014. – 94 с.
2. Розанов, А. К. Быстрый алгоритм анализа словоформ естественного языка с трехуровневой моделью словаря начальных форм / А. К. Розанов // Cloud of science. – 2016. – № 1. – С. 115–124.
3. Осминин, П. Г. Построение модели реферирования и аннотирования научно-технических текстов, ориентированной на автоматический перевод : автореф. дис. … канд. филол. наук: 10.02.21 / П. Г. Осминин. – Челябинск : ФГБОУ ВПО ЮУрГУ, 2016. – 108 с.

ПРИМЕНЕНИЕ МЕТОДОВ BIG DATA В ПРОГРАММНОМ КОМПЛЕКСЕ КОНСОЛИДАЦИИ ИНФОРМАЦИИ ОБ АВИАРЕЙСАХ

Ю. Ю. Пашковская

Учреждение образования «Гомельский государственный технический университет имени П. О. Сухого», Республика Беларусь

Научный руководитель Т. А. Трохова

В настоящее время одной из актуальных задач в области получения информации об авиасообщениях является задача автоматизации процесса консолидации больших объемов данных, включающая возможность получения выборок различного типа по запросу пользователя. Тема доклада посвящена решению этой актуальной проблемы. Разработанная система предназначена для формирования хранилища больших объемов данных с применением методологий и технологий Big Data. Применение этой системы позволит пользователям получать достоверную информацию об авиаперелетах по разным категориям запросов в короткое время.

Программный комплекс разработан на платформе Spring boot. Одним из отличительных моментов платформы Spring boot является применение паттерна MVC.

Концепция паттерна MVC предполагает разделение приложения на три компонента:

– модель (model);
– представление (view);
– контроллер (controller).

В качестве модели выбраны java классы, описывающие все поля, которые будут располагаться в базе данных. Так называемые pojo объекты для описания методов GET и SET; будет использоваться библиотека генерации кода Lombok. Она позволяет существенно уменьшить объем кода.

Представлением являются страницы html, которые содержат код пользовательского интерфейса в основном на языке html с thymeleaf.

В контроллере предполагается обработка запросов пользователя. Для работы с базой данных реализован репозиторий. Он необходим для того, чтобы работа с ней

происходила отдельно от выполнения контроллера. Также это способствует уменьшению количества кода.

Таким образом, проектируемая система состоит из одного веб-приложения, фреймворка распределенной обработки данных, базы данных и ее репозитория. Общая схема программного комплекса представлена на рис. 1.

Рис. 1. Схема программного комплекса

Spark – это проект Apache, который позиционируется как инструмент для «молниеносных кластерных вычислений». Проект разрабатывается процветающим свободным сообществом, в настоящий момент является наиболее активным из проектов Apache.

Spark предоставляет быструю и универсальную платформу для обработки данных. По сравнению с Hadoop Spark ускоряет работу программ в памяти более чем в 100 раз, а на диске – более чем в 10 раз.

Spark имеет следующие ключевые черты:

– в настоящее время предоставляет API для Scala, Java и Python, также готовится поддержка других языков (например, R);

– хорошо интегрируется с экосистемой Hadoop и источниками данных (HDFS, Amazon S3, Hive, HBase, Cassandra и т. д.);

– может работать на кластерах под управлением Hadoop YARN или Apache Mesos, а также функционировать в автономном режиме.

Ядро Spark дополняется набором мощных высокоуровневых библиотек, которые бесшовно стыкуются с ним в рамках того же приложения. В настоящее время к таким библиотекам относятся SparkSQL, Spark Streaming, MLlib (для машинного обучения) и GraphX. Сейчас также разрабатываются другие библиотеки и расширения Spark.

Ядро Spark – это базовое ядро для крупномасштабной параллельной и распределенной обработки данных. Ядро отвечает за:

– управление памятью и восстановление после отказов;

– планирование, распределение и отслеживание заданий в кластере;

– взаимодействие с системами хранения данных

В Spark вводится концепция RDD – неизменяемая отказоустойчивая распределенная коллекция объектов, которые можно обрабатывать параллельно.

Так как данный программный комплекс написан на языке программирования Java, а также программный комплекс при сборке упаковывается в докер-контейнер,

то тем самым обеспечивается кроссплатформенность и легкий перенос на вышестоящее окружение. Также важно отметить, что контейнеризация приложения в современном мире является неотъемлемой частью большинства приложений. Докер-контейнер собирается во время сборки приложения, и затем он отправляется в хранилище образов DockerHub.

Обработка данных производится по расписанию (CRON-job), это значит, что в выбранное администратором время будет запускаться spark, который будет вычитывать данные с файловой системы (HDFS, S3). Архитектура приложения реализована с целью простого добавления новых коннекторов к различным источникам данных. Работа системы приведена на рис. 2.

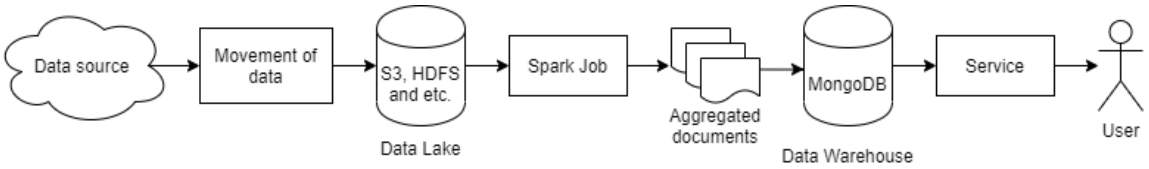

Рис. 2. Диаграмма последовательности работы системы

Компоненты обработки следующие:

- Data source – источник сырых данных.
- Movement of data – пока вручную, в будущем будет реализован сервис, основой которого будет трансфер данных с data source в data lake для последующей обработки данных.
- Data lake – холодное хранилище данных.
- Spark Job – распределенная обработка данных.
- MongoDB – документоориентированная база данных хранит подготовленные документы.
- Service – служит для аутификации и предпросмотра документов, а также для их экспорта.

Программный комплекс предназначен для построения статистики авиарейсов, основанной на различных критериях, выбранных пользователями. Программный комплекс должен обладать удобным интерфейсом и необходимым функционалом на основе методов обработки больших данных для выявления наиболее оптимального подхода разработки статистики авиарейсов.

Для реализации программного комплекса следует учесть разбиение функционала в зависимости от роли пользователя и дать каждому доступ к определенным функциям и ограничить доступ к иным. Программный комплекс предусматривает разделение на три роли:

– администратор;

– пользователь;

– пользователь с подпиской.

Каждый из пользователей должен быть зарегистрирован, чтобы иметь возможность использовать приложение. Если это новый пользователь, то он может зарегистрироваться самостоятельно.

Клиент должен иметь следующий функционал:

– авторизация в приложении;

– выбор необходимых критериев;

– просмотр статистики (ограничение в 100 записей);
– сохранение отчета построенной статистики;
– предварительный просмотр сохраняемых данных.

При покупке подписки клиент становится ее обладателем и имеет возможность просматривать более точную статистику, основанную на всех доступных записях. Клиент с подпиской должен иметь такой функционал, как:

– авторизация в приложении;
– выбор необходимых критериев;
– просмотр статистики, построенной на всех доступных данных;
– сохранение отчета построенной статистики;
– предварительный просмотр сохраняемых данных.

Администратор должен иметь следующий функционал:

– авторизация в приложении;
– назначение статуса клиента;
– изменение данных клиента;
– удаление клиента;
– добавление нового пользователя;
– просмотр статистики (ограничение в 100 записей);
– сохранение отчета построенной статистики;
– предварительный просмотр сохраняемых данных.

Программный комплекс создан с использованием технологий и различных фреймворков, обеспечивающих целостность и корректность введенных и отображаемых данных, и является надежным и кроссплатформенным решением. Веб-приложение как основная часть комплекса построена на базе паттерна MVC, что позволяет с легкостью масштабировать и сопровождать данное приложение.

ПРОГРАММНЫЙ КОМПЛЕКС АНАЛИЗА ПОЛЕТОВ АВИАРЕЙСОВ МЕТОДАМИ ОБРАБОТКИ БОЛЬШИХ ОБЪЕМОВ ДАННЫХ

Ю. Ю. Пашковская

Учреждение образования «Гомельский государственный технический университет имени П. О. Сухого», Республика Беларусь

Научный руководитель Т. А. Трохова

Цель разработки программного комплекса заключается в том, чтобы дать пользователю (пассажиру или авиакомпании) возможность быстро получить информацию о задержках авиарейсов, отмене рейсов, пассажиропотоке за любой период на любых направлениях и для любых авиакомпаний.

Программный комплекс состоит из таких компонентов как:

– база данных MongoDB;
– веб-приложение, разработанное с использованием паттерна MVC;
– хранилище данных S3;
– Spark-задача.

Обработка данных в программном комплексе выполняется следующим образом.

В системе данные копируются в хранилище данных S3, далее Spark-задача через определенный период, связанный с поступлением данных во входную систему, запускает обработку этих данных и записывает результат в mongodb, которое нахо-