

превосходство сверточной топологии над полносвязной, и влияние количества обучающих данных на эффективность распознавания изображений.

Список использованных источников

1. А. В. Кучуганов, Г. В. Лапинская Распознавание рукописных текстов, Ижевск 2006
2. 3 Gradient Based Learning Applied to Document Recognition, IEEE Press. – 1998, Le Cunn Y., Bottou L., Haffner P. – Mode Access: <http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf>
3. База данных рукописных символов (цифр) MNIST. [Электронный ресурс] / – Режим доступа: <http://yann.lecun.com/exdb/mnist/index.html>
4. Дорогой Я.Ю. Архитектура обобщенных сверточных нейронных сетей, Киев, 2012.

ПРОГРАММНОЕ СРЕДСТВО ДЛЯ ЗАГРУЗКИ МЕДИЦИНСКИХ ЛОГИСТИЧЕСКИХ ДАННЫХ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Лепунова Е. А.

Глухова Л.А. – канд. техн. наук, доцент

Развитие электронного обмена данными (через Интернет или сети с добавленной стоимостью) позволяет компаниям связать товарные потоки с соответствующими информационными потоками, что облегчает равномерную работу цепей поставок. В логистике для медицинской промышленности существуют следующие участники цепи поставок: промышленность, логистика, медицинские учреждения. Предлагаемое в докладе программное средство относится к звену промышленности. Основной функцией данного программного средства является обеспечение возможности автоматизированной регистрации данных, гарантирующих высокое качество информации и прослеживаемости.

Можно выделить следующие принципы работы электронного обмена данными:

1. Производитель медицинского оборудования предоставляет список логистических единиц, которые будут импортироваться во внутреннюю систему логистической компании.
2. Все предоставленные требования специфицируются и стандартизируются.
3. Логистическая компания выдает производителю медицинского оборудования аутентификационные данные для FTP клиента и устанавливает на выделенном сервере необходимое программное обеспечение. Данное программное обеспечение включает сервис, отвечающий за обработку пришедших от клиента данных, и веб-сайт, предназначенный для контроля и управления процесса обработки и загрузки данных.
4. Производителю медицинского оборудования требуется экспортировать данные из внутренней ERP системы, преобразовать их в CSV формат и выслать файлы посредством FTP клиента.
5. Логистической компании требуется проверить данные согласно спецификации, импортировать данные во внутреннюю систему и предоставить клиенту отчет о проведенной загрузке.
6. В случае обнаружения сервисом ошибок данных, производитель медицинского оборудования обязуется исправить ошибки и повторно выслать данные. В случае возникновения технических неполадок, логистическая компания обязуется устранить неполадки, обеспечить нормальную работу программного обеспечения и импортировать данные во внутреннюю систему.

Основными преимуществами вышеописанной технологии электронного обмена данными являются: прямое снижение накладных расходов по ведению документооборота (объем первичных бумажных документов, трудозатраты персонала, курьерская служба, почтовые расходы, оплата услуг электросвязи и пр.); упрощение взаимодействия с клиентами; оптимизация и повышение эффективности бизнеса; надежность и конфиденциальность передаваемых данных; возможность интеграции новых логистических единиц.

Вышеназванные преимущества определяются следующими факторами: повышением скорости всего процесса оформления, обработки и прохождения документов (полностью автоматизированный процесс); снижением числа ошибок (исключение «человеческого» фактора) и снижение затрат на поиск и исправление ошибок; использованием защищенного протокола передачи данных (SFTP).

К основным недостаткам рассматриваемой технологии могут быть отнесены: обработка данных большими объемами (до 4 МВ); возможность взаимодействия только с одним форматом данных. Причинами проявления вышеприведенных недостатков являются: наличие усиленной технологии валидации данных, что увеличивает время обработки данных; возможность быстрого роста базы данных клиента.

Для создания рассматриваемого программного средства были использованы следующие технологии: платформа .NET, язык программирования C#, структурированный язык запросов SQL, платформа SSIS.

Основной сложностью при создании архитектуры программного средства стала необходимость предусмотреть механизм интеграции новых логистических единиц с минимальными модификациями. С учетом этого было принято решение, что часть логики программного обеспечения будет перенесена на уровень базы данных. Это означает, что при добавлении новой логистической единицы нет необходимости вносить изменения в рабочий код. Необходимо лишь добавить несколько новых записей в базу данных,

создать дополнительные таблицы и хранимые процедуры, а также создать несколько новых *ssis* пакетов. Таким образом, внесением изменений только на уровне данных, обеспечивается возможность управления логистическими единицами. В докладе рассматривается организация программного средства для загрузки медицинских логистических данных, позволяющая снизить накладные расходы по ведению документооборота, устранить поддержку ненужных документов, автоматизировать импорт новых логистических единиц. Рассматриваемое программное средство за счет использования протокола SFTP обеспечивает безопасную транспортировку данных, за счет наличия усиленной технологии валидации обеспечивает контроль дубликатов или некорректных данных.

Список использованных источников:

1. «Ассоциация автоматической идентификации «ЮНИСКАН / ГС1 РУС». Логистика для медицинской промышленности. [Электронный ресурс] — Режим доступа. — URL: <http://www.gs1ru.org/files/2721/StandartyGS1variant.pdf>.
2. «Мировой рынок систем электронного документооборота» [Электронный ресурс] — Режим доступа. — URL: <http://cifforum.ru/consulting/docflow/market/article1.8.200222.html> (Дата обращения 16.11.2012).

МОДЕЛИ РЕАЛИЗАЦИИ ПРОЦЕССА НЕПРЕРЫВНОЙ ИНТЕГРАЦИИ РАЗРАБАТЫВАЕМОГО ПРОГРАММНОГО СРЕДСТВА

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Агебян В.К.

Бахтизин В. В. — канд. техн. наук, доцент

Непрерывная интеграция — это практика разработки программных средств, которая заключается в автоматической и регулярной сборке и тестировании программных продуктов. Разрабатываемая модель позволяет оптимизировать время процесса интеграции для наиболее раннего выявления и устранения ошибок и противоречий.

В процессе разработки комплексных программных средств разработчики часто объединяют результаты своей работы друг с другом. Каждая интеграция проверяется автоматической сборкой и тестированием для скорейшего определения интеграционных проблем. Данный подход существенно повышает эффективность процесса разработки и качество программного средства. Усовершенствование процессов интеграции и тестирования позволяет своевременно выявлять и устранять дефекты уже в процессе разработки, а не эксплуатации или сопровождения.

Основной целью непрерывной интеграции является обнаружение проблем настолько быстро, насколько это возможно для скорейшего выявления ошибок. Если сборка системы и выполнение тестов занимают много времени, теряются основные преимущества в использовании данного подхода. Для комплексных программных средств наибольшее время при интеграции занимает, как правило, этап тестирования.

Для оптимизации времени выполнения тестов предлагается распределенная модель реализации системы непрерывной интеграции. Используется архитектура ведущих/ведомых серверов. Для работы системы непрерывной интеграции программного продукта выделяется кластер. Для синхронизации работы кластера необходим выделенный сервер, который является единым центром управления.

В задачи распределенной системы входят:

- получение исходного кода текущей версии продукта из системы контроля версии;
- сборка текущей версии продукта;
- выполнение тестов;
- формирование отчетов.

Сборку текущей версии продукта необходимо выполнить на каждом сервере из кластера. Далее исходный набор тестов делится на группы. Группы тестов добавляются в глобальную очередь выполнения. Тесты распределяются по группам таким образом, чтобы суммарное время выполнения набора тестов в каждой группе максимально совпадало. За счет этого достигается оптимальная загруженность выделенных серверов. Оценка времени выполнения каждого теста формируется как экспоненциально взвешенное скользящее среднее на основе времени предыдущих выполнений данного теста:

$$EMA_n = a * p_n + (1 - a) * EMA_{n-1},$$

где p_n — время последнего успешного выполнения теста, a — сглаживающая константа.

Таким образом, необходимо хранить статистику о времени выполнения тестов. Каждый сервер в кластере после завершения выполнения набора тестов запрашивает центральный сервер и выполняет следующий набор тестов из очереди. После выполнения всех тестов статистика о результатах тестирования отправляется центральному серверу для формирования результирующего отчета.

Для реализации серверной логики используется язык программирования Python. Графический