Вычислительный комплекс для контроля качества распознавания структурированных документов*

П.В.Безматерных[1], Е.Л.Плискин[2] и В.В.Фарсобина[2]

[1] ООО «Смарт Энджинс Сервис», г. Москва, Россия
[2] Федеральное государственное учреждение «Федеральный исследовательский центр "Информатика и управление" Российской академии наук», г. Москва, Россия

Аннотация. На сегодняшний день вычислительный эксперимент остаётся ежедневной рутинной процедурой при разработке программного обеспечения (ПО) с использованием методов обучения машин (МОМ). Известный под названием «непрерывной интеграции» подход к разработке ПО является естественным выбором при создании программ МО и, со своей стороны, предполагает частую централизованную сборку программы и выполнение стендовых испытаний. При этом генерируется большой объём результатов испытаний, которые должны быть оперативно доступны разработчикам для анализа ошибок и сравнения версий ПО. Авторами статьи разработана архитектура системы автоматического контроля качества программы распознавания структурированных документов, включая сбор, хранение и отображение результатов стендовых испытаний. Результаты испытаний ПО записываются в базу данных. Стендирование ПО может выполняться на виртуальных серверах под управлением различных операционных систем (ОС). Для устойчивости веб-сервер и база данных физически отделены от сборочного сервера. Веб-технологии используются как для автоматической загрузки результатов испытаний в БД, так и для обслуживания запросов пользователей.

Ключевые слова: компьютерный эксперимент, методы обучения машин, реляционные базы данных, веб-технологии, регрессионное тестирование, непрерывная интеграция, контроль качества.

DOI 10.14357/20718632180208

Введение

«Непрерывная интеграция» [1] есть подход к разработке программного обеспечения (ПО), предполагающий использование централизованной системы контроля версий (СКВ), в которую разработчики ежедневно вливают новый код. При каждом изменении рабочей версии ПО выполняется автоматизированная сборка программы и выполнение стендовых испытаний на заданном наборе исходных данных. Непрерывная интеграция направлена на более ранее обнаружение и устранение ошибок. При тестировании ПО ошибки могут обнаруживаться как в новых, так и в старых модулях, в которые не вносились изменения в данной версии. В этом случае говорят о *регрессионных ошибках*. В литературе предложены способы сокращения объёма регрессионного тестирования, предполагающие выборочное тестирование изменённых и зависимых от них модулей [2].

При разработке ПО с использованием методов обучения машин (МОМ) вычислительные эксперименты являются неотъемлемой частью цикла разработки. Типичный эксперимент заключается в применении программы МО к множеству исходных файлов, которое разработчики на своём жаргоне называют «стендом», а эксперимент называют «прогоном стенда». Если говорить о ПО для распознавания документов, то стенд может содержать сканы,

*Работа выполнена при финансовой поддержке РФФИ, гранты № 15-29-06086 и №16-29-12925.

фотографии и видеофрагменты (клипы) с изображениями документов, а также файлы разметки в таком машиночитаемом формате, как, например, JSON. Файл разметки содержит информацию об изображённом на фотографии (клипе, скане) документе, включая расположение полей (окон) документа и тексты полей, правильно распознать которые в идеале и должна тестируемая программа. Например, файл разметки для каждого включённого в стенд изображения документа может содержать текстовые значения и координаты на изображении документа полей «серия», «номер», «фамилия», «имя», «отчество» и т.д. Результаты стендовых испытаний версии ПО основаны на автоматическом сравнении фактических выходных данных программы распознавания с «идеальными» значениями полей в разметке стенда. Это сравнение выполняется тестирующей программой, отличной от тестируемого ПО.

Искусственные нейронные сети (ИНС, нейросети) широко применяются [3] для обучения машин, включая задачи распознавания. Нейросетевые технологии применяются тогда, когда формализация задачи трудна или невозможна, поскольку неизвестен характер связи между входом и выходом. При обучении нейронной сети различают её «весовые параметры» и «гиперпараметры». Весовые параметры автоматически вычисляются при помощи алгоритмов обучения нейросети на основе *тренировочной выборки* изображений. Напротив, гиперпараметры подбираются разработчиками эвристически. Подбор гиперпараметров требует множественных экспериментов с использованием обученной нейронной сети для распознавания *тестовой выборки*. Гиперпараметры фигурируют также и в других моделях обучения машин, помимо нейронных сетей.

Таким образом, при разработке программы распознавания по технологии «непрерывной интеграции», вообще говоря, необходимость частых вычислительных экспериментов определяется двумя взаимосвязанными факторами: новыми версиями программы и необходимостью подбора гиперпараметров для моделей обучения машин.

В литературе описаны различные виды ПО для обработки результатов научных экспериментов [6, 7]. Нам, однако, не известны описания архитектуры системы автоматического стендирования, сбора, хранения и отображения результатов распознавания структурированных документов. Данная работа стремится восполнить этот пробел.

1. Вычислительный комплекс

На Рис. 1 показана предлагаемая схема вычислительного комплекса для контроля качества программы распознавания. Комплекс состоит из двух физически отдельных серверов, связь между которыми осуществляется через Интернет. В правой части Рис. 1 показан сборочный сервер, а в левой ча-

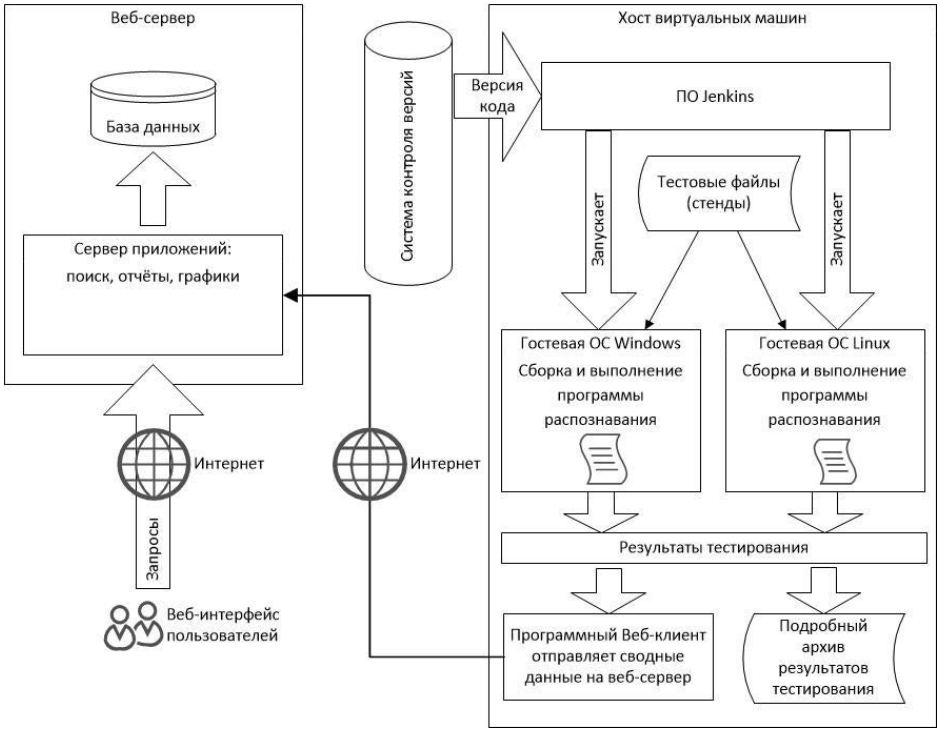

Рис. 1. Архитектура вычислительного комплекса

сти веб-сервер, выполняющий также функции сервера базы данных (БД). На сборочном сервере функционирует ПО Jenkins [4, 5], которое управляет двумя виртуальными машинами с гостевыми ОС Windows и Linux и дирижирует периодическими процессами сборки и стендовых испытаний версий ПО, код которых извлекается из системы контроля версий. Также на сборочном сервере располагаются исходные тестовые файлы (стенды) и архив результатов испытаний. В то время как подробные результаты распознавания остаются на сборочном сервере, сводные данные о каждом прогоне стендов поступают в находящийся на сборочном сервере программный веб-клиент, который пересылает их через Интернет на сервер приложений.

Отправку сводных данных на веб-сервер целесообразно осуществлять асинхронно, для устойчивости к возможным перерывам в Интернет-связи. Перерывы в Интернет-связи не должны затруднять работу сборочного сервера. Для этого у программного веб-клиента имеется файловый кэш, где временно хранятся данные до отправки на веб-сервер. На веб-сервере функционирует программный сервер приложений, который принимает поступающие сводные данные о результатах испытаний и заносит их в БД и обрабатывает запросы пользователей.

2. Структурированные документы

За последнее десятилетие технологии распознавания текстовых документов из актуальной научной задачи превратились в стандартные надёжные функции информационных систем [7, 8]. Однако внедрение систем массового ввода сложноструктурированных документов, как правило, не обходится без настройки, а то и серьёзной доработки ПО для распознавания документов. Это связано, прежде всего, с большим разнообразием накопленных бюрократических приёмов оформления бумажных документов, распознавание которых может иной раз затруднить даже естественный человеческий интеллект, а тем более программы с элементами «искусственного интеллекта» (ИИ).

Не до конца решены и проблемы, возникающие в процессе использования в задачах ввода документов фото- и видеосъёмки с использованием стационарных и мобильных малоразмерных цифровых видео камер, к которым можно, в первую очередь, отнести веб-камеры и камеры мобильных устройств [10, 11]. Недостаточное разрешение камеры, вариации условий освещения и позиционирования документов относительно камеры при съёмке создают трудности и вынуждают разработчиков ПО распознавания использовать любые доступные уникальные особенности того или иного внедренческого проекта для настройки-доработки своих программ. В итоге мы снова приходим к вычислительным экспериментам, на материале конкретного заказчика или целевой группы потенциальных заказчиков.

К структурированным документам относятся различные бумажные анкеты, бланки, удостоверения, формы, таблицы. Они характеризуются наличием «полей» или «окон», которые могут располагаться в документе как на фиксированных относительно границ документа позициях, так и «плавать», поодиночке или образуя устойчивые группы. Для поиска полей документа могут использоваться координаты, линии разметки, статические тексты, шрифтовые особенности и т.п. Количество и расположение полей может меняться от документа к документу.

Для целей описания архитектуры системы контроля качества мы предположим, что множество тестовых документов разбито на «типы документов», а для каждого типа документов определено множество полей, которые характеризуются текстовыми метками. Например, в некотором типе документов могут быть выделены поля «серия», «номер», «фамилия», «имя», «отчество» и т.д. Если однотипные поля или группы полей в документе могут повторяться неизвестное заранее число раз, например, в виде таблицы, то будем снабжать метки полей цифровыми суффиксами, такими как, например, номер строки таблицы.

Вообще говоря, в одном стенде могут присутствовать различные типы документов. Одна и та же метка может присваиваться полям в различных типах документов. Это имеет смысл, если для распознавания данного поля в различных типах документов используются одинаковые алгоритмы.

3. Компоненты программы распознавания

Распознавание структурированных документов есть сложный процесс, который выполняется поэтапно. На каждом этапе выполняется некоторая программная компонента, которая получает на вход результаты предыдущего этапа, а свои результаты передаёт следующему этапу. Типичный конвейер распознавания документов может включать следующие этапы [9, 12, 13].

- Определение границ документа. На этом этапе может выполняться выпрямление наклонного или изогнутого изображения, например, снятого веб-камерой.
- Определение типа документа.
- Детектирование фотографии, печати, подписи – нетекстовых элементов документа.
- Выделение текстовых полей.
- Выпрямление наклонного шрифта.

- Нарезка полей на символы текста, так называемая «сегментация». Нарезка основана на нахождении промежутков между символами текста. Промежутки выявляются в проекции на линию строки. В результате этого этапа формируются «ячейки» – координаты ограничивающих символы текста прямоугольников.
- Распознавание символов текста. В результате этого этапа для каждой символьной ячейки формируется список «альтернатив». Каждая альтернатива включает вариант текстового символа, такой как букву, цифру или знак пунктуации, а также оценку вероятности. Например, список альтернатив может содержать варианты букв «ш» и «щ» с различной вероятностью. Список альтернатив сортируется по убыванию вероятности.
- Контекстная корректировка текста полей. На этом этапе учитываются возможные сочетания букв в зависимости от языка. Например, в русском языке не встречаются сочетания букв «кщ» или «щп». В результате списки альтернатив корректируются: некоторые варианты могут исключаться, а вероятность других вариантов может повышаться или понижаться, что влияет на сортировку списка альтернатив.
- Словарная корректировка текста полей. Словарь выбирается в зависимости от поля документа. Например, словарь фамилий, словарь имён или словарь отчеств.

Для контроля качества отдельных компонентов программы распознавания могут применяться такие стенды, в разметке которых заданы результаты одного или нескольких начальных этапов [14]. Например, для проверки компоненты, выполняющей выпрямление наклонного шрифта, в разметке стенда могут быть заданы идеальные координаты полей. В общем случае в разметке стенда могут присутствовать следующие элементы: тип документа; координаты документа на изображении; координаты устойчивых групп полей («зон»), таких как фамилия, имя, отчество; координаты нетекстовых элементов документа (фотографии, печати, подписи); угол наклона и другие свойства шрифта для каждого поля; тексты полей. При использовании методов обучения машин целесообразно разделить каждый стенд на открытую разработчикам и закрытую от них части, во избежание «переобучения» алгоритмов, то есть подгонки под конкретную совокупность тестовых документов.

4. Результаты стендовых испытаний

Сводные данные об одном прогоне стенда, отправляемые через Интернет со сборочного сервера серверу приложений для хранения в БД, могут включать следующие показатели.

1. Общие показатели прогона стенда:
   - Название проекта, номер версии ПО.
   - Название стенда, вид источника (сканы, фото, видеоклипы), операционная система, дата прогона, длительность прогона, среднее время распознавания документа.
   - Общий признак успешности прогона: не сломалась ли программа.
   - Общее количество входных документов, из них количество принятых и отвергнутых программой распознавания.
   - Среднее количество кадров видеоклипа, которые понадобились для распознавания документа.
   - Сведения о распознавании типа документов: количество документов с правильно/неправильно определённым типом документов.
   - Строка из нулей и единиц по числу документов стенда, отражающая правильное (1) или неправильное (0) *определение типа документов*. Такая строка позволяет обнаруживать случаи улучшения и ухудшения при сравнении с предыдущим прогоном того же стенда.
   - Ссылка для скачивания подробных результатов заданного прогона со сборочного сервера через Интернет, по запросу пользователя. Для скачивания данных со сборочного сервера на веб-сервер по запросу пользователя можно использовать протокол обмена файлами, такой как, например, SFTP.
2. Для каждого поля:
   - Строка из нулей и единиц по числу документов стенда, отражающая правильное (1) или неправильное (0) распознавание данного поля. Такая строка позволяет обнаруживать случаи улучшения и ухудшения при сравнении с предыдущим прогоном того же стенда.
3. Показатели обнаружения полей в документах, без учёта распознавания текста:
   - «False negatives» (FN) – количество пропущенных полей.
   - «False positives» (FP) – количество артефактов, ложно принятых за поле документа.
   - «True positives» (TP) – количество правильно найденных полей.

Revision	Changes, total items	Run time	Match	name	number	patronymic	surname
28561	Pos: 7099 Neg: 0 Itm: 2949	S: 129 Ms: 234	T.: 0.8769 Ch.: 0	Q: % Ch:	Q: 50.32% Ch: 0	Q: % Ch:	Q: % Ch:
28516	Pos: 23 Neg: 6983 Itm: 2949	S: 124 Ms: 237	T.: 0.8769 Ch.: 0	Q: % Ch:	Q: 50.32% Ch: -31	Q: % Ch:	Q: % Ch:
28445	Pos: 116 Neg: 19 Itm: 2949	S: 105 Ms: 204	T.: 0.8769 Ch.: 0	Q: 71.43% Ch: 0	Q: 51.37% Ch: +116-19	Q: 66.67% Ch: 0	Q: 68.09% Ch: 0
28335	Pos: 177 Neg: 43 Itm: 2949	S: 106 Ms: 204	T.: 0.8769 Ch.: 0	Q: 71.43% Ch: +36-10	Q: 48.08% Ch: +59-3	Q: 66.67% Ch: +33-13	Q: 68.09% Ch: +16-3

Рис. 2. Список прогонов с результатами стендовых испытаний
Pos – улучшения, Neg – ухудшения

Цветом показаны улучшения и ухудшения по сравнению с предыдущими ревизиями ПО: зелёный – чистое улучшение качества, без ухудшений в документах; жёлтый – в документах больше улучшений, но есть и ухудшения; сиреневый – в документах не меньше ухудшений, чем улучшений; красный – чистое ухудшение качества, без улучшений в документах.

- «True negatives» (TN) – количество правильно определённых случаев отсутствия поля в документе.
- P = (TP + FN) – фактическое количество полей в документах.
- N = (TN + FP) – количество недостающих полей в документах.
- «Sensitivity» или «True positive rate» (TPR) или «Recall» – чувствительность = (TP/P).
- «Specificity» (SPC) или «True negative rate» (TNR) – специфичность = (TN/N).
- «Precision» или «Positive predictive value» (PPV) – позитивная предсказательная ценность = (TP / (TP + FP)).
- «Negative predictive value» (NPV) – негативная предсказательная ценность = (TN / (TN + FN)).
- «Accuracy» (ACC) – точность = ((TP + TN)/P + N).

4. Показатели распознавания текста полей документов, могут вычисляться как для каждого поля в отдельности, так и суммарно по всем полям:

- Качество – доля документов, в которых текст данного поля распознан верно.
- Показатели, вычисляемые по сравнению с предыдущим прогоном того же стенда, т.е. по сравнению с предыдущей подвергнутой испытаниям ревизией ПО:
  - Количество документов, в которых распознавание поля *улучшилось*.
  - Количество документов, в которых распознавание поля *ухудшилось*.
  - Направление изменения качества распознавания текстов, для возможной цветовой кодировки,. Рис. 2: (+2) – чистое улучшение качества, без ухудшений в документах, зелёный цвет; (+1) – в документах больше улучшений, но есть и ухудшения, жёлтый цвет; 0 – качество осталось без изменений; (-1) – в документах не меньше ухудшений, чем улучшений, сиреневый цвет; (-2) – чистое ухудшение качества, без улучшений в документах, красный цвет.

5. Показатели распознавания типа документов, которые вычисляются по сравнению с предыдущим прогоном того же стенда:

- Количество документов, в которых распознавание типа документов *улучшилось*.
- Количество документов, в которых распознавание типа документов *ухудшилось*.
- Направление изменения качества распознавания типа документов, от (+2) до (-2). Вычисляется аналогично направлению качества распознавания текстов полей, как описано в п.4 выше.

6. Показатель качества сегментации – интегральная характеристика чистоты промежутков между символами текста, от 0 до 1.

7. Координаты обнаруженных в документе фотографии, печати, подписи.

5. База данных

База данных, в которой хранятся результаты стендовых испытаний программы распознавания документов, может иметь структуру, показанную на Рис. 3. Центральным информационным объектом можно считать «прогон». Прогон соответствует од-

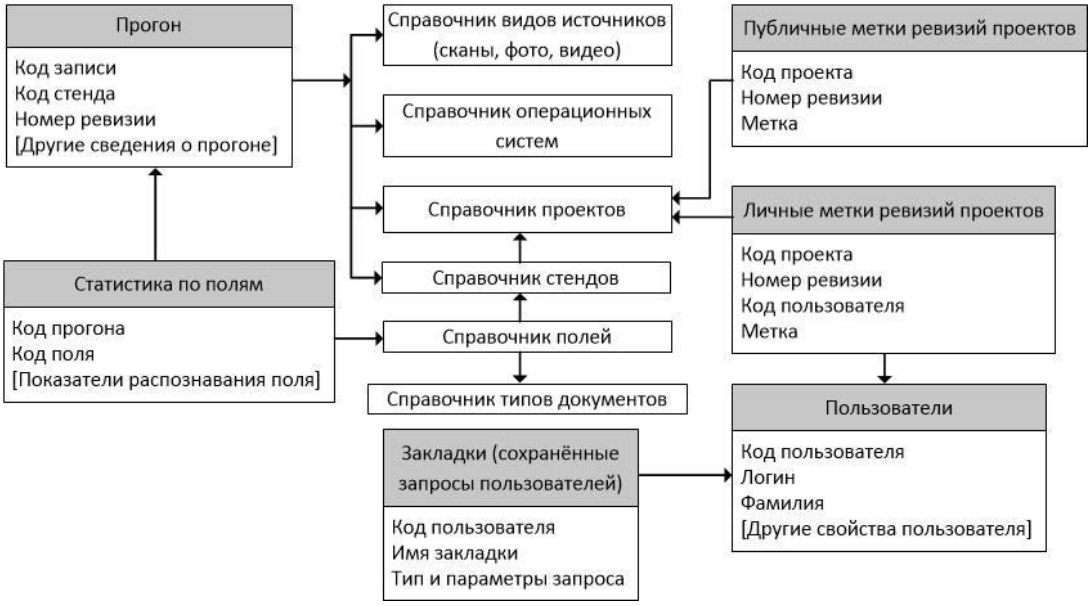

Рис. 3. Схема базы данных с результатами стендовых испытаний

Стрелками показаны связи между таблицами типа N:1
(первичный ключ целевой записи может содержаться в нескольких записях исходной таблицы)

ному вычислительному эксперименту, в ходе которого обрабатывается заданный стенд. Результаты прогона образуют пакет данных, который единовременно передаётся со сборочного сервера на сервер приложений и записывается в БД.

Все ссылки на Рис. 3 однозначные. Прогон ссылается на проект, стенд, операционную систему и вид источников (сканы, фото, видео). Стенд, в свою очередь, ссылается на «проект». Под проектом может пониматься как базовый набор алгоритмов распознавания, так и конкретный внедренческий проект настройки-доработки ПО. Стенд включает в себя набор тестовых исходных изображений для распознавания, в которых могут присутствовать один или несколько различных типов документов. Статистика по полям привязана к прогону и к справочнику полей. Справочник полей привязан, с одной стороны, к стенду, а с другой стороны, к справочнику типов документов. Таким образом, в центре Рис. 3 образуется иерархия (проект – стенд – поле).

Номера ревизий генерируются системой контроля версий автоматически. Некоторые «интересные» или «важные» ревизии могут вручную помечаться разработчиками текстовыми метками. Метки ревизий могут быть как публичными, видными всем пользователям, так и личными, видными одному пользователю. В БД могут также храниться «закладки» веб-интерфейса – сохранённые запросы пользователей.

6. Веб-интерфейс пользователей

Веб-интерфейс пользователей может выполнять следующие виды запросов к хранящимся в БД сводным данным о результатах стендовых испытаний различных версий программы распознавания структурированных документов.

- Список стендов, с возможной выборкой по проекту, по ревизии, по дате прогона. Для каждого стенда могут отображаться номер ревизии и дата последнего прогона.
- Страница поиска прогонов. Обязательные параметры: проект, стенд, вид источника, ОС. Дополнительные фильтры: скрыть прогоны без улучшений и без ухудшений; минимальный номер ревизии; минимальная дата прогона. Результаты поиска могут представляться либо в виде списка прогонов, либо в виде графика. В таблице прогонов могут отображаться характеристики прогона; статистика по полям; а также количество улучшений и ухудшений по сравнению с предыдущим прогоном того же стенда (т.е. с предыдущей испытанной ревизией ПО), для каждого поля и суммарно по всем полям. На графике (Рис. 4 по оси значений может отображаться качество распознавания для всех полей стенда, а по оси времени – номера ревизий и даты прогонов. Таблица прогонов может быть скачана в формате CSV.

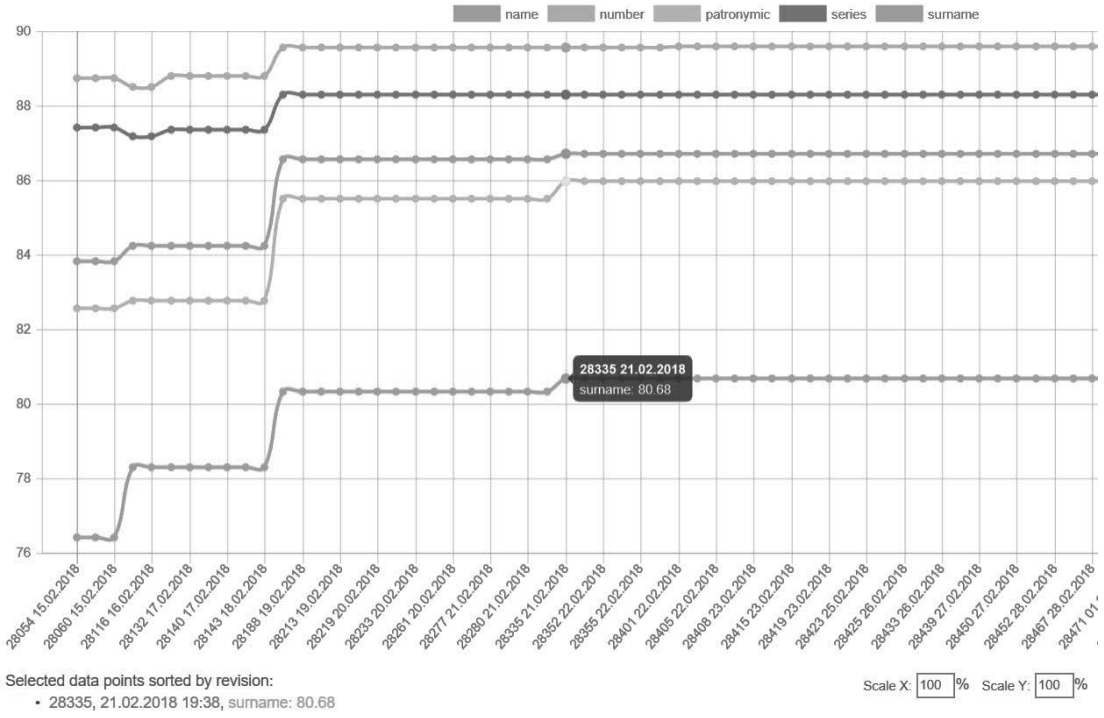

Рис. 4. График качества распознавания документов в веб-интерфейсе пользователей

name - фамилия, number - номер, patronymic - отчество, series - серия, surname - имя. По оси времени показаны номера ревизий и даты прогонов. По оси значений процент верно распознанных текстов полей

- Сравнение произвольного множества ревизий с заданной «базовой» ревизией. Обязательные параметры: проект, стенд, вид источника, ОС. В множество ревизий для сравнения могут добавляться интервалы номеров ревизий и интервалы по дате прогонов. Дополнительные фильтры при выборе ревизий для сравнения: наличие метки ревизии; наличие улучшений или ухудшений. Результаты сравнения ревизий можно отображать в виде таблицы, подобно результатам поиска прогонов, но улучшения и ухудшения вычислять не по сравнению с предыдущим прогоном того же стенда, а по сравнению с заданной «базовой» ревизией.
- Сравнение результатов испытаний заданной ревизии проекта на различных ОС для выявления нештатной зависимости работы ПО от операционной системы.
- Список ревизий заданного проекта, с возможной фильтрацией по номеру ревизии и дате прогона. Для каждой ревизии можно отображать дату прогона; суммарное по всем стендам количество изменений (улучшений и ухудшений) по сравнению с предыдущим прогоном того же стенда; количество стендов, на которых наблюдались изменения; общее количество стендов; количество документов в стендах с изменениями и во всех стендах.
- Список закладок (сохранённых запросов) пользователя.
- Страница подробной информации о прогоне, включая статистику по полям; возможность скачивания подробных результатов прогона со сборочного сервера; возможность редактирования публичной и личной меток ревизии.
- Страница подробной информации о ревизии, включая список стендов, на которых испытывалась эта ревизия, с возможностью скрыть стенды без изменений. Для каждого стенда можно отобразить количество улучшений и ухудшений, а также статистику по полям. На этой странице также можно предусмотреть возможность редактирования публичной и личной меток данной ревизии.

Литература

1. Duvall PM, Glover A, Matyas S. Continuous integration. Addison-Wesley Professional; 2007.
2. Elbaum, S., Rothermel, G. and Penix, J., 2014, November. Techniques for improving regression testing in continuous integration development environments. In Proceedings of the

22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering (pp. 235-245). ACM.
3. Касторнова В. А., Можаева М. Г. Искусственные нейронные сети как современные средства информатизации. Информационная среда образования и науки, 2012, №1 (7).
4. Сервер автоматизации сборки и тестирования ПО Jenkins: https://jenkins.io/.
5. Smart, J.F. «Jenkins: The Definitive Guide». O'Reilly Media, Inc. 2011. ISBN: 1449305350 9781449305352.
6. Косенко Д.В., Воронова Л.И., Воронов В.И. Разработка программного обеспечения для обработки сложноструктурированных данных научного эксперимента. Вестник Нижневартовского государственного университета, 2014, № 3.
7. Полевой Д.В., Самойлов, О.С. Задача контроля качества при создании и развитии систем оптического распознавания печатного текста. Технологии программирования и хранения данных/Труды Института системного анализа РАН 45 (2009): 251-259.
8. Полевой Д.В. Актуальные задачи создания систем массового ввода с использованием оптического распознавания для преобразования сложно структурированных бумажных документов в гибридных информационных системах // Системный анализ и информационные технологии: тр. Четвертой междунар. конф. (Абзаково, Россия, 17-23 августа 2011 г.): в 2 т. Т.2. Челябинск: Изд-во Челяб. гос. ун-та, 2011. С. 192-195.
9. Арлазаров В.Л., Куратов П.А., Славин, О.А. Распознавание строк печатных текстов. Сб. трудов ИСА РАН «Методы и средства работы с документами». М.: Эдиториал УРСС, 2000, с.31-51.
10. Bulatov K., Arlazarov V., Chernov T., Slavin O. and Nikolaev D. Smart IDReader: Document recognition in video stream // The 14th IAPR International Conference on Document Analysis and Recognition (ICDAR 2017), Workshops and Tutorials: November 9-12, Kyoto, Japan, 2017 – p. 39-44. ISSN: 2379-2140 http://ieeexplore.ieee.org/document/8270294/, doi: 10.1109/ICDAR.2017.347.
11. Полевой Д., Булатов К., Скорюкина Н., Чернов Т., Арлазаров В.В., Шешкус А.В. Ключевые аспекты распознавания документов с использованием малоразмерных цифровых камер. Вестник РФФИ, 2016, № 4 (92), С. 97-108.
12. Skoryukina N., Chernov T., Bulatov K., Nikolaev D. and Arlazarov V.L. Snapscreen: TV-stream frame search with projectively distorted and noisy query. Proc. SPIE 10341, Ninth International Conference on Machine Vision (ICMV 2016), 103410Y, pp. 1-5. doi:10.1117/12.2268735.
13. В.Л. Арлазаров, А. Марченко, Д. Шоломов. Накопительные контексты в задаче распознавания. Труды ИСА РАН, 2014, Т. 64. №4, с. 64-72.
14. Будаковский М.В., Михайлов А.А. Проблемы формализации разметки графического образа документа. Труды ИСА РАН, 2014, Т.64. № 4. С. 84-88.

Безматерных Павел Владимирович. ООО «Смарт Энджинс Сервис», г. Москва, Россия. Научный сотрудник-программист. Количество печатных работ: 7. Область научных интересов: системы обработки документов. E-mail: bezmpavel@gmail.com

Плискин Евгений Львович. Федеральное государственное учреждение «Федеральный исследовательский центр "Информатика и управление" Российской академии наук», г. Москва, Россия. Ведущий научный сотрудник, кандидат технических наук. Количество печатных работ: 20. Область научных интересов: автоматизированные информационные системы. E-mail: pliskin@isa.ru

Фарсобина Вера Викторовна. Федеральное государственное учреждение «Федеральный исследовательский центр "Информатика и управление" Российской академии наук», г. Москва, Россия. Научный сотрудник. Количество печатных работ: 23. Область научных интересов: распознавание образов, вычислительные эксперименты. E-mail: farsobina@isa.ru

Information system for structured documents OCR quality control

P.V.Bezmaternyh[1], E.L.Pliskin[2] and V.V.Farsobina[2]

[1] Smart Engines Service, Moscow, Russia
[2] Federal Research Center “Computer Science and Control” of Russian Academy of Sciences, Moscow, Russia

To date, the computational experiment remains a daily routine procedure during development of machine learning (ML) based software, such as optical character recognition (OCR). Well-known approach of «continuous integration» (CI) is a natural choice for the development of ML software. CI involves frequent centralized program builds and execution of bench tests. This generates a large amount of test results, which should be readily available to developers for error analysis and software version comparison. This article suggests the architecture of the automatic quality control system for the structured documents OCR, including collection, storage and display of bench test results. The results of all software tests are loaded into the database. Builds and bench tests can execute on virtual servers running various operating systems (OS). For stability, the web

server and database use different hardware from the build server. Web technologies are used both for automatic uploading of test results to the database and for servicing user queries.
Keywords: computer experiment, machine learning, data processing, web applications, regression testing, continuous integration, quality control.

DOI 10.14357/20718632180208

Reference

1. Duvall PM, Glover A, Matyas S. Continuous integration. Addison-Wesley Professional; 2007.
2. Elbaum, S., Rothermel, G. and Penix, J., 2014, November. Techniques for improving regression testing in continuous integration development environments. In Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering (pp. 235-245). ACM.
3. Kastornova V.A., Mozhaeva M.G. Artificial neural networks as modern means of informatization. Information environment of education and science, 2012, №1 (7).
4. Open source automation server Jenkins, https://jenkins.io/.
5. Smart, J.F. «Jenkins: The Definitive Guide». O'Reilly Media, Inc. 2011. ISBN: 1449305350 9781449305352.
6. Kosenko D.V., Voronova L.I. and Voronov V.I., 2014. Development of software for processing complex-structured data of a scientific experiment. Bulletin of Nizhnevartovsk State University, № 3.
7. Polevoy D.V. and Samoilov O.S., 2009. Quality control in development of systems for optical recognition of printed text. Technologies of programming and data storage / Proceedings of the Institute of System Analysis of the Russian Academy of Sciences, 45 (2009): 251-259.
8. Polevoy D.V., 2011. Actual problems of creating mass data input systems using optical recognition for the transformation of complex structured paper documents in hybrid information systems // System analysis and information technologies: Proc. Fourth Intnl. Conf. (Abzakovo, Russia, August 17-23, 2011), vol.2, Chelyabinsk: Publishing house Chelyab. state University, 2011. p. 192-195.
9. Arlazarov V.L., Kuratov P.A. and Slavin O.A., 2000. Recognition of lines of printed texts. Collected works of ISA RAS «Methods and tools for working with documents». Moscow: Editorial URSS, 2000, p.31-51.
10. Bulatov K., Arlazarov V., Chernov T., Slavin O. and Nikolaev D., 2017. Smart IDReader: Document recognition in video stream // The 14th IAPR International Conference on Document Analysis and Recognition (ICDAR 2017), Workshops and Tutorials: November 9-12, Kyoto, Japan, 2017 – p. 39-44. ISSN: 2379-2140 http://ieeexplore.ieee.org/document/8270294/, doi: 10.1109/ICDAR.2017.347.
11. Polevoy D., Bulatov K., Skoryukina N., Chernov T., Arlazarov V.V. and Sheshkus A.V., 2016. Key aspects of document recognition using small-sized digital cameras. Vestnik RFBR, 2016, No. 4 (92), pp. 97-108.
12. Skoryukina N., Chernov T., Bulatov K., Nikolaev D. and Arlazarov V.L., 2016. Snapscreen: TV-stream frame search with projectively distorted and noisy query. Proc. SPIE 10341, Ninth International Conference on Machine Vision (ICMV 2016), 103410Y, pp. 1-5. doi:10.1117/12.2268735.
13. Arlazarov V.L., Marchenko A. and Sholomov D., 2014. Cumulative contexts in the recognition problem. Proceedings of the ISA RAS, 2014, Vol. 64. No. 4, p. 64-72.
14. Budakovsky M.V. and Mikhailov A.A., 2014. The problems of formalizing markup of a graphic image of a document. Proceedings of ISA RAS, 2014, vol.64, № 4, p. 84-88.

Pavel V. Bezmaternykh. Smart Engines Service, Moscow, Russia. Researcher and developer. Number of publications: 7. Research interests: document recognition systems. E-mail: bezmpavel@gmail.com

Eugene L. Pliskin. PhD. Federal Research Center "Computer Science and Control" of Russian Academy of Sciences, Moscow, Russia. Leading Researcher. Number of publications: 20. Research interests: automated information systems. E-mail: pliskin@isa.ru

Vera V. Farsobina. Federal Research Center "Computer Science and Control" of Russian Academy of Sciences, Moscow, Russia. Researcher. Number of publications: 23. Research interests: pattern recognition, computational experiments. E-mail: farsobina@isa.ru