АВТОМАТИЧЕСКИЙ ПОИСК ОПОРНЫХ ЭЛЕМЕНТОВ НА ДОКУМЕНТАХ ПОЛУЖЕСТКОЙ СТРУКТУРЫ

М.О. ЛАНИН
аспирант кафедры распознавания изображений и обработки текста, факультет инноваций и высоких технологий, Московский физико-технический институт (государственный университет); программист, ООО «Аби Продакшн»
Адрес: 141700, Московская обл., г. Долгопрудный, Институтский пер., д. 9
E-mail: mike.lanin@gmail.com

Статья посвящена проблеме потокового извлечения данных из документов полужесткой структуры, для которых слабо применимы методы сплошного оптического распознавания символов. Для облегчения процесса создания структурных описаний таких документов широко используются методы машинного обучения. Тем не менее, существующие решения по-прежнему достаточно сложны для конечного пользователя, поскольку требуют ручного описания элементов структуры документа, не имеющих прямого отношения к извлекаемым данным.

В работе рассматривается возможный подход к описанию изображений документов переменной структуры, используемый в системе потокового ввода ABBYY FlexiCapture, а также метод автоматического построения такой структурной модели по разметке всех элементов структуры. Подробно описывается алгоритм автоматического поиска опорных элементов по пользовательской разметке извлекаемых данных, позволяющий значительно облегчить с точки зрения пользователя процесс создания структурной модели документа ABBYY FlexiCapture. Интеграция описанной технологии обучения на этапе верификации извлекаемых данных позволяет инкрементально улучшать структурную модель документа, при этом все, что требуется от конечного пользователя – исправлять неверно найденные в процессе ввода регионы извлекаемых полей. Также в статье описан метод и результат оценки эффективности предложенного подхода. Описанный способ поиска опорных элементов показал свою применимость на реальных платежных документах ряда немецких поставщиков: 89,3% счетов могут быть обработаны без ошибок при минимальном участии пользователя, при этом верно извлечены данные из 97,8% полей.

Ключевые слова: система потокового ввода, документы полужесткой структуры, структурное описание изображения документа, опорные элементы, реперы, поля, разметка полей, методы машинного обучения, частотный словарь.

1. Ведение

По мере все большего распространения распределенных информационных систем и систем электронного документооборота набирают популярность и системы потокового ввода данных. На вход таких систем попадает большой массив документов различной природы, поэтому обработка всего объема данных вручную затруднительна, затратна по времени и практически невозможна. Построение эффективной системы потокового ввода документов связано с решением целого ряда задач, таких как распознавание текста, классификация изображений, маршрутизация и извлечение данных из документов.

В данной статье внимание уделено проблеме автоматического извлечения данных из изображений документов. Даже документы одного типа зачастую имеют полужесткую структуру, то есть расположение однотипных данных на них варьируется от одного экземпляра к другому. Использование

трафаретных форм с явным указанием регионов извлекаемых данных и дальнейшим распознаванием текста методами оптического распознавания символов не позволяет корректно извлекать информацию из таких документов. Для извлечения информации из документов нежесткой логической и графической структуры используется специализированная структурная модель документа [1; 2]. Большую трудность в использовании таких моделей представляет необходимость прямого указания геометрических отношений между элементами документа [3; 4], создания тематических словарей [2; 3], а также ручного описания некоторых элементов структуры [2; 4]. Все это обусловливает высокие требования к квалификации создателей структурных описаний и большие временные затраты на создание эффективных описаний даже для документов простой структуры.

Для упрощения процесса создания структурных описаний документов используются методы машинного обучения. Показал свою эффективность метод автоматического построения структурного описания документа с использованием пользовательской разметки всех элементов [5]. Однако, этот подход хоть и значительно упрощает процесс создания структурных описаний, но не решает проблему в целом. От пользователя по-прежнему требуется вручную указывать набор статических элементов (подписи, заголовки и т.д.), с этим связаны определенные трудности. Во-первых, пользователь не всегда может хорошо определить, является ли строка «хорошим» опорным элементом (репером), надежно локализующим поле. Во-вторых, возникает ряд проблем, если предполагается, что процесс создания структурного описания может быть распределен по времени или пространству, так как набор размеченных реперных элементов должен совпадать для всех документов обучающей выборки. Следующим шагом в упрощении процесса создания структурного описания является автоматизация поиска опорных элементов документа на основе разметки извлекаемых данных (полей). Такой подход до минимума снижает требовательность к квалификации пользователя, поскольку требует лишь указания положения полей на изображении документа, а также позволяет внедрить систему обучения на этапе верификации извлекаемых данных и инкрементально улучшать структурное описание в процессе ввода.

В работе рассматривается подход к автоматическому поиску реперных элементов для дальнейшего построения структурного описания в рамках модели, используемой в системе ABBYY FlexiCapture. Применимость предложенного метода была проверена на тестовом пакете из 622 реальных немецких счетов от 10 поставщиков. Метод показал свою эффективность, как часть процесса инкрементального обучения структурного описания в процессе ввода.

2. Модель структурного описания документов

Методы автоматического поиска опорных элементов рассматриваются в контексте модели структурного описания, используемой в системе потокового ввода ABBYY FlexiCapture. Структурное описание документа представляет собой дерево типизированных элементов, определяющих формат и содержимое некоторых частей документа. Всего доступно 18 типов элементов для различных типов данных, таких как статический текст, дата, денежная сумма, число, телефон, регулярное выражение и т.д. Для каждого из элементов задается набор свойств, определяющих как содержимое всего документа (обязательные и запрещенные элементы, минимальное и максимальное количество подэлементов в группе, количество повторений и т.д.), так и содержимое самих элементов (варианты текста, минимальное и максимальное значение, формат, количество символов, количество строк и т.д.). Кроме того, для каждого элемента могут быть заданы ограничения области поиска – набор полуплоскостей, в которых элемент может быть найден. Ограничения области поиска могут быть заданы относительно границ изображения либо относительно расположения других, найденных ранее, элементов.

На основе заданных свойств и областей поиска определяется качество гипотезы о том, что распознанное слово или множество слов является соответствующим элементом. Нарушение каждого из правил приводит к наложению определенного штрафа, размер которого зависит от степени нарушения и пользовательских настроек. В процессе анализа документа строится дерево гипотез: для каждой из гипотез расположения текущего элемента выдвигается множество гипотез расположения следующих элементов. Качество цепочки гипотез рассчитывается как произведение качеств каждой из гипотез этой цепочки. Результатом анализа является цепочка гипотез наилучшего качества, если

качество такой цепочки превосходит предельное минимальное значение; в противном случае считается, что документ не соответствует заданному структурному описанию. В общем случае логика построения структурного описания заключается в определении статического «каркаса» документа и последующей локализации извлекаемых данных относительно его.

3. Метод построения структурного описания по пользовательской разметке

Подробное описание метода автоматического построения структурного описания документа по пользовательской разметке выходит за рамки данной статьи, однако для общего понимания рассмотрим краткое описание используемого алгоритма. Для задачи построения структурного описания не рассматривается проблема классификации документов: считается, что структурное описание создается для документов схожей структуры (счета выделенного поставщика, анкеты определенного клиента и т.д.). В случае, когда на вход поступают документы различной структуры, подразумевается использование внешнего классификатора и нескольких структурных описаний для каждого из типов обрабатываемых документов.

С учетом вышеуказанных предположений и описанной ранее модели структурного описания, задача автоматического создания структурного описания по пользовательской разметке всех элементов сводится к двум подзадачам: определение типа и настройка свойств каждого из элементов и построение геометрических отношений между элементами. Набор свойств элемента зависит от его типа, и алгоритм их настройки не будет рассматриваться подробно. В общем случае, свойства задаются как наиболее строгие разумные ограничения, включающие все возможные варианты обучающей выборки.

Для построения геометрических ограничений области поиска для каждого документа строится матрица отношений между элементами типа «выше-ниже» и «левее-правее», после чего результирующая матрица отношений получается как пересечение всего множества матриц. Отступы ограничений от границ элементов также задаются как наиболее строгие ограничения, включающие все возможные варианты взаимного расположения элементов на документах набора обучения.

4. Поиск опорных элементов с использованием пользовательской разметки полей

Вкратце, поиск опорных элементов состоит из следующих этапов:

1. построение частотного словаря для документов обучающей выборки;

2. выбор слов-кандидатов в реперные элементы по частотности;

3. слияние геометрически близких слов-кандидатов в строки для каждого из документов;

4. сопоставление реперов на документах обучающей выборки.

Для поиска опорных элементов для всех документов обучающей выборки строится частотный словарь с учетом возможных ошибок распознавания: слова считаются вариантами одного слова, если сводятся друг к другу не превосходящим порогового значения количеством элементарных операций, к которым относятся вставка, удаление и замена символа.

На основе частотных словарей выбираются слова-кандидаты, при этом для оценки качества слова w используется мера

$$q(w) = \frac{\sum_i q(d_i, w)}{|D|},$$

где $|D|$ – количество документов в наборе обучения, а $q(d_i, w)$ – качество слова w на документе d_i, равное $\max_j(q(d_i, w_j))$ – максимальному качеству конкретного экземпляра слова на документе. Качество каждого конкретного экземпляра слова вычисляется в соответствии с форматом содержимого, форматированием, геометрическими размерами и расположением текста. Так, бонус получают слова форматно похожие на заголовки, URL, подписи и т.п. статические элементы документа. Слова, входящие в мелкий и сплошной текст, получают меньшее качество. Также штрафуются слова по формату похожие на численные значения, даты, пунктуаторы и т.п.

Дополнительно отсекаются слова с малой частотностью, по мере

$$df(w) = log(1 + \frac{|d_i \ni w|}{|D|}),$$

где $|d_i \ni w|$ – количество документов обучающей выборки, в которых встречается слово w, а $|D|$ – общее количество документов в наборе обучения.

На этом этапе не используется мера для отсечения слов, часто встречающихся на одном экземпляре документа, поскольку частоупотребительное в контексте документа слово может быть частью менее частотной строки, являющейся хорошим опорным элементом. Например, частотные на платежных документах слова «invoice» и «date» образуют надежный репер-подпись к полю «invoice date».

На следующем этапе геометрически близкие слова-кандидаты сливаются в строку на каждом из документов набора обучения. Список опорных элементов получается путем сопоставления полученных строк-кандидатов для каждого из документов. Начальный список реперов формируется на основе строк-кандидатов, найденных на первом документе, дальнейший процесс сопоставления происходит итеративно следующим образом:

1. Для каждого следующего документа строится двудольный граф, вершинами которого являются строки-кандидаты и имеющиеся на данном этапе реперы, соответственно. Вес каждого ребра для репера r и строки-кандидата s рассчитывается по разметке множества полей F как

$$W(r,s) = T(r,s)\cdot(\max_i L(r,s,f_i) - k\frac{\sum_{j=1}^{N} L(r,s,f_j)}{|F|}),$$

где $T(r, s)$ – степень совпадения текстового содержания r и s, $L(r, s, f_i)$ – степень совпадения относительного расположения поля f_i относительно r на коллекции обработанных документов и относительно s на текущем документе, $|F|$ – общее количество полей, а k – константа.

2. Из графа удаляются все ребра с качеством ниже порогового, после чего ищется максимальное паросочетание, определяющее принадлежность строки кандидата к соответствующему реперу. Строки, не имеющие парного репера, добавляются как новый реперный элемент.

3. Для всех реперных элементов обновляются данные об относительном расположении полей с учетом последнего обработанного документа.

На последнем этапе, после окончания процесса сопоставления, из полученного списка реперных элементов удаляются элементы, представленные на малом относительном количестве документов обучающей выборки.

5. Методика и результат тестирования

Оценка эффективности метода автоматического поиска реперных элементов сама по себе затруднительна, поскольку, во-первых, не всегда можно объективно определить, насколько качественным репером является конкретная строка, во-вторых, зачастую для построения качественного структурного описания документа не обязательно использовать все «надежные» статические элементы, а достаточно ограничиться каким-то разумным минимумом. В связи с этим характеристики полноты и точности относительно некоторой эталонной разметки реперов не являются приемлемыми показателями качества вне контекста дальнейшего использования найденных опорных элементов.

Предложенный подход тестируется в связке с описанным ранее методом автоматического создания структурных описаний, как первый шаг инкрементального обучения структурного описания в процессе ввода: сперва по пользовательской разметке полей определяется расположение опорных элементов, затем объединение разметки реперов и полей используется для создания очередной версии структурного описания. Эффективность такого подхода сравнивается с автоматическим созданием структурного описания с использованием вручную указанных опорных элементов: для документов обучающей выборки оператор указывает расположение подписей к извлекаемым данным, а также без использования опорных элементов, исключительно по положению полей.

В общем виде сценарий тестирования выглядит следующим образом:

1. Для начала обработки документов, пользователь вручную указывает расположение извлекаемых данных на первых трех изображениях документов данного типа.

2. По окончании разметки автоматически создается первая версия структурного описания, используемого в дальнейшем для извлечения данных из документов.

3. Если в процессе потоковой обработки какие-то поля нашлись неверно, пользователь вручную исправляет расположение полей.

4. В случае, когда ошибки обработки не связаны с дефектами изображения (проблемы распознавания, перекосы и т.д.), пользователь добавляет проблемный экземпляр документа в набор обучения.

5. На основе пользовательской разметки полей по всему набору обучения создается очередная версия структурного описания, которая используется для дальнейшей обработки документов.

Для проверки применимости предложенного подхода использовался пакет, содержащий 622 реальных счета от 10 немецких поставщиков. Таким образом, в процессе ввода участвовало 592 документа, поскольку первые три изображения документов каждого из поставщиков использовались для создания исходной версии структурного описания.

Таблица 1.

Результаты обработки документов при обучении без использования реперов

Количество ошибок	Количество документов	Доля документов
0	0	0%
1	0	0%
2	6	1,0%
3	62	10,5%
4	58	9,8%
5	134	22,6%
6	260	43,9%
7	72	12,2%

Результаты эксперимента, приведенные в *табл. 1*, показывают, что метод обучения без использования статических элементов не пригоден для потокового ввода. Так, ни один документ не был введен без ошибок, менее 3 ошибок было допущено всего на 1% документов, а для более, чем половины документов потока выделено не более одного поля.

Таблица 2.

Результаты извлечения данных полей при обучении без использования реперов

Имя поля	Количество полей в пакете	Корректно найдено полей	Доля найденных полей
Получатель	592	161	27,2%
Адрес получателя	590	64	10,8%
Номер счета	574	191	33,2%
Дата	592	372	62,8%
Сумма без налога	580	10	1,7%
Сумма налога	492	57	11,6%
Сумма	588	28	4,8%

Из *табл. 2* видно, что более-менее приемлемый результат извлечения данных без использования опорных элементов достигается только для достаточно жестко позиционированных полей выделенного формата (дата). Жестко располагающиеся поля более свободного формата выделяются сильно хуже (получатель, номер счета), а качество выделения полей с плавающей позицией (сумма счета) или переменного размера (адрес получателя) нельзя назвать хоть сколько-нибудь приемлемым.

Таблица 3.

Сравнение результатов обработки документов при автоматическом поиске и ручном указании опорных элементов

Количество ошибок	Количество документов		Доля документов	
	авто	ручной	Авто	Ручной
0	529	418	89,3%	70,6%
1	46	122	8,6%	20,6%
2	9	33	1,5%	5,6%
3	3	10	0,5%	1,7%
4	2	5	0,3%	0,8%
5	0	1	0%	0,1%
6	3	3	0,5%	0,5%

Таблица 4.

Сравнение результатов извлечения данных полей при автоматическом поиске и ручном указании опорных элементов

Имя поля	Количество полей в пакете	Корректно найдено полей		Найдено лишних полей	
		авто	ручной	авто	ручной
Получатель	592	577	555	0	0
Адрес получателя	590	571	467	2	0
Номер счета	574	551	536	4	4
Дата	592	585	572	0	0
Сумма без налога	580	570	570	8	12
Сумма налога	492	485	478	2	2
Сумма	588	580	575	0	0

бл. 3 и *4* приведено сравнение результатов ввода с инкрементальным обучением структурного описания при использовании автоматически найденных («авто») и указанных оператором вручную («ручной») опорных элементов. Описанный метод показал свою эффективность: 89,3% документов были обработаны без ошибок, а ошибка более чем в два поля была допущена менее, чем на 1,5% документов, при этом успешно было найдено 3919 из 4008 полей (97,8%). Более того, в контексте описанного инкрементального обучения структурной модели, предложенный подход оказался эффективнее, чем ручное указание оператором позиций подписей к извлекаемым полям.

6. Выводы

Метд показал свою эффективность и позволил минимизировать участие пользователя в процессе создания структурного описания. Результаты тестового эксперимента показали, что пользователь может получить структурное описание достаточно высокого качества, просто указывая расположение извлекаемых данных. Тем не менее, по-прежнему требуется поддержка пользователя для фильтрации дефектных изображений, что требует определенной, хоть и не высокой, квалификации. В идеале, система должна уметь автоматически принимать решение, добавлять ли очередное изображение в обучающую выборку.

Также не стоит забывать, что при разработке метода использовалось достаточно сильное предположение о том, что документы различной природы и структуры могут быть наверняка разделены внешним классификатором. Однако, само по себе создание такого классификатора – достаточно сложная и трудоемкая задача, требующая отдельного исследования. ■

Литература

1. Зуев К.А. Система идентификации структуры печатных документов. Дисс. канд. тех. наук. М.: МГУЛ, 1999.
2. Hamza H., Belaid A., Belaid Y., Chaudhuri B. An end-to-end administrative document analysis system // Proceedings of the Eighth International Workshop on Document Analysis Systems (DAS 2008), Nara, Japan, September 16-19, 2008. P. 175-182.
3. Ishitani Y. Model-based information extraction method tolerant of OCR errors for document images // Proceedings of the Sixth International Conference on Document Analysis and Recognition (ICDAR), Seattle, Washington, USA, September 10-13, 2001. P. 908-915.
4. Peanho C., Stagni H., da Silva F. Semantic information extraction from scanned images of complex documents // Applied Intelligence. December 2012. Vol. 37, No. 5. P. 543-557.
5. Голубев С.В. Распознавание структурированных документов на основе машинного обучения // Бизнес-информатика. 2011. № 2. С. 48-55.

AUTOMATIC DETECTION OF REFERENCE ELEMENTS ON SEMI-STRUCTURED DOCUMENT IMAGES

Mikhail LANIN
Post-graduate student, Department of Images Recognition and Text Processing, Faculty of Innovations and High Technologies, Moscow Institute of Physics and Technology (State University); Software engineer, ABBYY Production
Address: 9, Institutskiy per., Dolgoprudny, Moscow Region, 141700, Russian Federation
E-mail: mike.lanin@gmail.com

The paper deals with automatic data extraction from semi-structured documents. The through optical character recognition methods are slightly applicable for this kind of input. To simplify the process to create structural descriptions of such documents machine learning methods are widely used, however, current solutions are still complicated for end-users, because these require manual description of document structure elements, which are not directly relevant to date to be extracted.

The article presents a possible approach to describe variable structure document images used in document data capture system called ABBYY FlexiCapture and a method of automatic model creation based on layout of all structure elements. The paper provides a detailed description of an algorithm for automatic detection of reference elements based on user layout of data to be extracted that enables to facilitate dramatically the process of building of a structured model of an ABBYY FlexiCapture document from the user perspective. Integration of this technology at the data extraction validation stage enables to incrementally improve the structural model of a document, as it requires a user only to correct localization of wrongly found data being extracted. Finally, the paper describes a method to assess robustness of the proposed approach and test results. The described method involving detection of reference elements has shown its effectiveness in processing actual payment documents of a number of German suppliers: 89.3% of invoiced can be treated with no faults with minimum user intervention; furthermore, the data had been extracted correctly from 97.8% of fields.

Key words: data capture, semi-structured documents, structural description of document, reference elements, reference points, fields, fields layout, machine learning, frequency list.

References

1. Zuev K.A. (1999) *Sistema identifikatsii struktury pechatnykh dokumentov* (diss. kand. tekh. nauk) [Printed documents structure identification system (PhD Thesis)]. Moscow: MGUL. (in Russian)
2. Hamza H., Belaid A., Belaid Y., Chaudhuri B. (2008) An end-to-end administrative document analysis system. Proceedings of Proceedings of the *Eighth International Workshop on Document Analysis Systems (DAS 2008)*, Nara, Japan, September 16-19, 2008. P. 175-182.
3. Ishitani Y. (2001) Model-based information extraction method tolerant of OCR errors for document images. Proceedings of the *Sixth International Conference on Document Analysis and Recognition (ICDAR)*, Seattle, Washington, USA, September 10-13, 2001, pp. 908–915.
4. Peanho C., Stagni H., da Silva F. (2012) Semantic information extraction from scanned images of complex documents. *Applied Intelligence*, vol. 37, no. 5, pp. 543-557.
5. Golubev S.V. (2011) Raspoznavanie strukturirovannykh dokumentov na osnove mashinnogo obucheniya [Structured documents recognition based on machine learning]. *Business Informatics*, no. 2 (16), pp. 48-55. (in Russian)