

АДАПТАЦИЯ МАТЕМАТИЧЕСКОЙ МОДЕЛИ НЕЙРОННОЙ СЕТИ ДЛЯ СИСТЕМЫ РАСПОЗНАВАНИЯ СЛИТНОГО РУКОПИСНОГО ТЕКСТА

Е.В. Долгова, Д.С. Курушин

*Пермский государственный технический университет
614990, Пермь, Комсомольский пр., 29*

Аннотация. В статье предлагается подход к адаптации нейросетевых моделей для создания OCR-систем, рассчитанных на работу со слитным неограниченным рукописным текстом. Подход основан на отказе от распознавания слитных символов и переходу к распознаванию отдельных штрихов, из которых затем собираются символы и/или слова текста. Рассматриваемый подход позволяет существенно понизить размерность нейронных сетей, используемых в OCR-системах, что приведет к повышению их производительности и качества распознавания.

Ключевые слова: нейронная сеть, математическая модель, python, распознавание, вектор, граф, языковая модель.

Технологии распознавания рукописного текста развиваются уже в течении более чем 15 лет и рукописный текст используется в различных сферах, включая образование, здравоохранение, банковское дело, страхование, в правительственных и других структурах. Существует два глобальных подхода к проблеме распознавания рукописного текста — распознавания «на лету» (т.н. on-line распознавание) и распознавание написанного ранее текста, представленного в виде растрового изображения (т.н. off-line распознавание).

Хотя эти подходы имеют много общего (что обусловлено однотипными объектами анализа), задача распознавания текста, непосредственно вводимого оператором решается достаточно успешно, и эта технология применяется в различных устройствах. С другой стороны, распознавание оцифрованных рукописей до сих пор представляет собой трудно разрешимую задачу.

В настоящее время многие исследователи, (напр. см. [4]) полагают, что распознавание рукописного текста может выполняться по следующему алгоритму:

1. Выделение слов или словосочетаний (основываясь на промежутках между словами);
2. Сегментация текста на элементы (символы, штрихи...);
3. Распознавание элементов;
4. Генерация выходного текста.

В данной работе мы рассмотрим один из возможных подходов к распознаванию штрихов, составляющих рукописный текст.

Распознавание штрихов, представленных наборами точек (в растровом формате) может выполняться любым традиционным способом, например могут использоваться нейронные сети различных конфигураций (см. напр. [2]). Однако, использование нейронных сетей в таком контексте приводит или к необходимости нормализовывать размеры изображения. При нормализации изображения к размерности входного вектора сети неизбежно возникают нелинейные искажения, что ухудшает качество распознавания, или требует увеличения размерности сети. Второй вариант хоть и не приводит к

искажениям входного сигнала, тоже не лишен недостатков. Любая нейронная сеть большой размерности требует пропорционально большого количества элементов в обучающей выборке [1]. Подготовка таких массивов данных сложна технически и может порождать ошибки, обусловленные человеческим фактором.

Слитный рукописный текст характеризуется относительно большим различием в начертаниях одинаковых символов. В зависимости от ближайшего окружения один и тот же символ может писаться по-разному. Отличия в начертании возникают и при изменении скорости письма, геометрической формы бумажного носителя.

Другой сложностью, возникающей при распознавании слитного текста является необходимость сегментации слова на символы перед распознаванием. Задача сегментации слитного рукописного текста может не иметь единственного формального решения. Таким образом, ошибочная сегментация слова на символы может приводить к падению качества распознавания отдельных символов. Действительно, попытка распознавания частей символов, случайным образом объединенных друг с другом при ошибочной сегментации, будет приводить к фактически случайным результатам распознавания.

Решением проблемы сегментации символов может быть увеличение размерности входного вектора нейронной сети таким образом, чтобы подавать на ее вход не отдельные символы, а слова целиком. Таким образом, возникает необходимость повышения размерности выходного вектора до, как минимум, количества слов в словаре системы. Однако, вариативность написания слов значительно выше вариативности написания символов. Действительно, пусть A среднестатистическое количество вариативных начертаний отдельного символа некоторого алфавита a ; пусть N среднестатистическая длина слова, тогда количество вариантов написания слова с использованием алфавита a можно грубо оценить как $M = N^{L_a \cdot A}$. На самом деле вариантов написания слова еще больше, т.к. могут видоизменяться не только символы, но и соединительные линии между ними. Для русского языка это число составит [3]:

$$M = 5.28^{33.5} \approx 1.7 \cdot 10^{119}, \quad (1)$$

при условии существования 5 вариантов написания каждого символа.

Конечно, в реальности число вариантов написания может быть меньшим, так, в этой-же работе, дается оценка, показывающая что для понимания 80% текстов достаточно знания 5000 слов, что, с учетом вариантов написания, даст оценку $M \approx 4.5 \cdot 10^{97}$.

Дополнительная сложность растрового распознавания слов заключается в необходимости нормализации исходных данных перед их подачей на вход нейронных сетей. Очевидно, что размерность входного вектора сети должна быть выше, чем в случае с распознаванием отдельных символов. Поскольку длина слова в языке колеблется от 1 до 10 – 20 символов, размерность входного вектора должна быть достаточной для помещения в него самых длинных слов без существенных потерь данных при нормализации. Для распознавания рукопечатных символов достаточно области 16·16 пикселей, что соответствует 256 элементам входного вектора.

Для распознавания целых слов имеет смысл использовать область пропорционально большего размера (по горизонтали), например 160·16 или 320·16 пикселей. Это соответствует от 2560 до 5120 элементов входного вектора. При этом число элементов вы-

ходного вектора сети должно соответствовать количеству слов в словаре системы (т.е. 5 – 50 тыс, в зависимости от задач системы).

Таким образом, число нейронов входного слоя оказывается порядка 5 тыс, выходного слоя – десятки тысяч. Такая конфигурация сети требует соответственного количества нейронов на внутренних слоях. И, хотя, размерность сети может быть существенно сокращена с использованием соответствующих методов (например двоичного кодирования выходного сигнала), очевидно, что такой подход требует значительных трудозатрат при подготовке сети к работе и существенных вычислительных ресурсов для обучения сети.

Для устранения этой проблемы можно использовать в качестве входного сигнала нейронной сети не растровое изображение, а его векторное представление. Под векторным представлением растрового изображения будем понимать такой нагруженный граф, у которого нагрузка вершин — пары координат (x, y) соответствующих узловых точек изображения. За узловые точки будем принимать точки соединения изображения линий в растровом изображении. В зависимости от выбираемой стратегии изображение может рассматриваться как состоящее из отрезков прямых, или дуг. Изображения, состоящие из дуг могут более точно соответствовать растровому оригиналу, но сложнее для последующего анализа.

В настоящей работе используется алгоритм *potrace*, представляющий растр набором замкнутых многоугольников или т.н. *безъугольников* — замкнутых фигур стороны которых заданы кривыми Безье. Как было отмечено выше, для дальнейшего анализа с нашей точки зрения многоугольники представляют больший интерес.

Традиционно, нейронные сети принимают в качестве входного вектора такие вектора \vec{x}_i , элементы которых принадлежат интервалам $[0,1]$ или $[-1,1]$. Следовательно, для решения задачи распознавания элементов рукописи нам необходимо описать функцию преобразования векторных представлений элементов рукописи в форму, пригодную для использования нейронной сетью. Сам факт возможности классификации элементов рукописи нейронной сетью мы будем считать доказанным.

Из документации на *potrace* [5] нам известно следующее: *potrace* конструирует направленный граф следующим образом: пусть p — точка с целочисленными координатами (соответствует 4-м пикселям изображения), такую точку *potrace* считает вершиной (*vertex*) и обозначать как v (или w), если эти четыре пикселя имеют различия по цвету. *Potrace* полагает, что между вершинами v и w существует грань (*edge*, e), если

$$E(v, w) = 1, \quad (2)$$

где E — Евклидово расстояние и если отрезок VW отделяет пиксель черного цвета от пикселя белого цвета, таким образом, что черный пиксель остается слева от условного направления движения. Продолжая двигаться таким образом от вершины к вершине мы получаем направленный граф G . Граф представляет собой путь (*path*, P) $\{v, K, v_n\}$, такой что:

$$\exists P, \exists e_{i \dots n}(v_i, v_{i-1}) = 1, \wedge e_i \neq e_j, \forall i, j \leq n \quad (3)$$

Путь P называется замкнутым, если $v_n = v_0$. Используя ряд методик, описанных в документации на potrace [5], potrace преобразует растровое изображение в набор замкнутых путей P_i . Замкнутый путь P_i соответствует некоторому элементу, обозначим его W_j . Индексы отличаются, т.к. некоторые пути могут соответствовать незначимым элементам, или оставшемуся после фильтрации шуму.

Таким образом, задача классификации путей P_i в элементы W_j может быть сформулирована следующим образом: найти такую классифицирующую функцию, что:

$$F(P_i) = \begin{cases} W_j, P_i \in W \\ \emptyset, P_i \notin W \end{cases}, \quad (4)$$

где W — множество известных рукописных элементов (задача определения такого множества является задачей лингвистического исследования и не включена в настоящую работу, хотя и выполнена автором). Будем считать, что F — многослойная нейронная сеть, тогда нам необходимо дополнительно преобразовать замкнутый путь P таким образом, чтобы каждый его элемент v_i принадлежал диапазону $[-1, 1]$.

Элементы пути P представляют собой вершины, т.е. вектора координат $\{x, y\}$. Эта форма представления информации удобна для отображения на экране, но нерациональна с точки зрения распознавания формы пути. Форму пути можно описать, разбив его на известное количество незамкнутых путей P' равной длины и рассмотрев угол поворота α_i при переходе $P'_i \rightarrow P'_{i+1}$. Чтобы отфильтровать возможные дефекты формы штриха, вызванные толщиной линии и (это главное) существенно упростить путь, перейдем от замкнутого пути P к незамкнутому P^m , представляющему собой среднюю линию. Для этого, для каждого пути P_i выполняем:

$$P_i^m = f^m(P_i), \quad (5)$$

где f^m — функция скелетизации контура. Определим ее:

$$\begin{aligned} f^m(P) &= f^m(v_0, \dots, v_n) = \{v_0^m, \dots, v_s^m\} \\ v_k^m &= V^m(v_i, v_{i+1}), \forall i, i < n, k = i/2, \end{aligned} \quad (6)$$

где $V^m(v_1, v_2)$ — вычисление координат по формуле определения координат середины отрезка (мы полагаем, что вершины в пути отсортированы удобным для нас способом, иначе нужно применить очевидное правило сортировки вершин). Очевидно, что в зависимости от сложности исходных путей и их геометрических размеров количество вершин в них может отличаться. Соответственно оно будет отличаться и в путях P^m .

Известно, что размерность входного вектора (обычно x) нейронной сети постоянна для всей выборки, поэтому к входным данным применяют процедуру нормализации. В нашем случае нормализация по размерности также включает в себя нормализацию по значению, действительно, угол поворота α_i пути P^m в каждой точке лежит в диапазоне $(-\pi, \pi)$. Приведение диапазона $(-\pi, \pi)$ к $(-1, 1)$ может быть выполнено или σ -

функцией, принятой в теории нейронных сетей или обычным линейным преобразованием. Использование σ -функции предпочтительнее, т.к. она «растягивает» небольшие углы поворота и «сжимает» предельные, что соответствует физике процесса письма. Таким образом, мы будем использовать:

$$\sigma(\alpha_i) = \frac{1}{1 + e^{\frac{10}{\pi} \alpha_i}}. \quad (7)$$

Осталось рассчитать значения α_i .

Как сказано выше, путь P^m необходимо разбить на конечное постоянное число сегментов. Для этого определяем суммарную длину пути P^m как:

$$L = |P^m| = \sum_{i=0}^{n-1} \{v_i, v_{i+1}\} \quad (8)$$

определяем длину сегмента как: $l=L/N$, где N — необходимое число сегментов и, одновременно, размерность входного вектора нейронной сети.

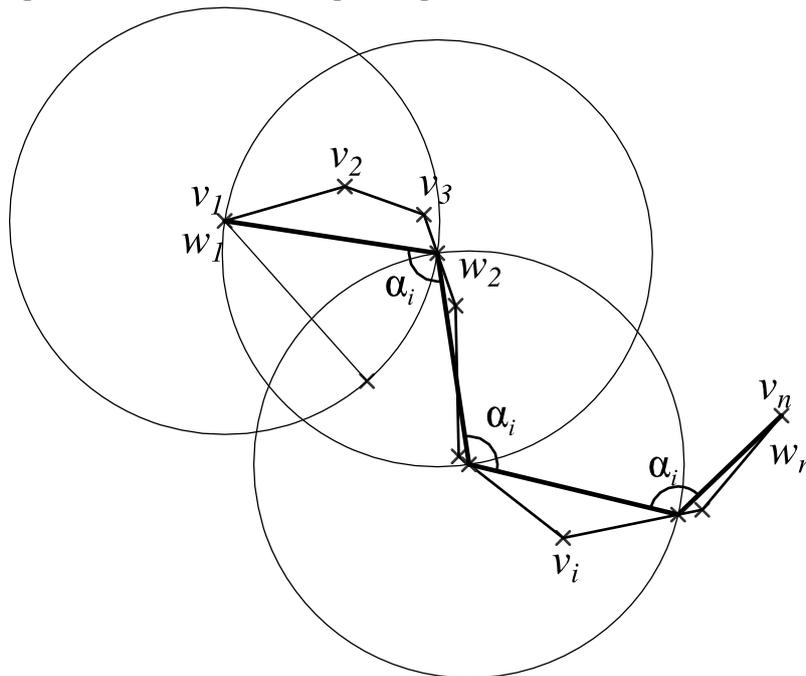


Рис.1. Геометрическая интерпретация разбиения штриха на сегменты

Для разбиения пути P^m на сегменты воспользуемся следующим способом (см. рис.1). Начиная с произвольно выбранного конца пути P^m , строится ряд окружностей радиуса l , так что центром первой является вершина v_1 , далее w_1 . Точка w_2 определяется методом поиска точки пересечения окружности и отрезка, с центром в этой точке строится новая окружность того же радиуса, определяющая вершину w_3 , и так далее, пока не будет достигнуто такое состояние, что все необработанные вершины лежат внутри очередной окружности. Тогда угол α_i есть угол между отрезками $w_i w_{i+1}$.

Таким образом, входной вектор нейронной сети \vec{x} рассчитывается следующим образом:

$$\begin{aligned} \vec{x} &= f^n(P^m) = f^n(\{v_1, \dots, v_n\}) = \{x_1, \dots, x_N\}, \\ x_j &= \sigma(\alpha_j), \\ \alpha_j &= f^a(w_{j-1}, w_j, w_{j+1}), j \in [2, N-1]. \end{aligned} \tag{9}$$

Вопросы подбора конфигурации и обучения нейронной сети лежат за рамками настоящего исследования, т.к. доказана принципиальная возможность решения любой задачи классификации [1]. Прделанные выше преобразования позволили выполнить процедуру нормализации изображения без потери содержательно-значимой информации, а также существенно сократить размерность сети, абстрагировавшись, одновременно, от начального угла поворота изображения (обычно представляющего сложность для сетей, распознающих растровое изображение).

Рассмотренная выше модель реализована на АЯП Python 2.6 и представлена в виде адаптации сети Кохонена (для предварительной автоматической классификации примитивов) и 3-х слойного персептрона. С использованием адаптированной сети Кохонена удалось выделить (на тестовых данных) 5 классов примитивов, использованных в дальнейшем как обучающая выборка для многослойного персептрона.

В результате было установлено что:

1. Использование векторного представления позволяет сократить размерность сети в десятки раз по сравнению с традиционным (растровым) методом анализа рукописи.
2. Адаптированная нейросетевая модель успешно распознает от 5 до 10 классов примитивов (на тестовых данных).

В дальнейшем предполагается переход к контекстно-зависимым нейронным сетям (напр. модифицированной сети Джордана), что позволит реализовать вариант РРМ-модели языка на дографемном уровне.

ЛИТЕРАТУРА

1. Долгова, Е.В., Курушин Д.С. Компьютерные нейросетевые технологии, Пермь, ПГТУ, 2008.
2. Мисюрёв, А.В. Использование искусственных нейронных сетей для распознавания рукопечатных символов, [Электронный документ] (<http://ocrai.narod.ru/hp.html>). Проверено 2010.12.20.
3. Шаров, С.А. Статистика слов в русском языке. [Электронный документ] (http://www.lingvisto.org/artikoloj/ru_stat.html). Проверено 12.02.2011.
4. Jaehwa, Park, Venu Govindaraju, and Sargur N. Srihari. Efficient word segmentation driven by unconstrained handwritten phrase recognition. In Proceedings of International Conference on Document Analysis and Recognition, pages 605–608, 1999.
5. Selinger, P. Potrace: a polygon-based tracing algorithm языке. [Электронный документ] (<http://potrace.sourceforge.net/potrace.pdf>). Проверено 12.02.2011.

A NEURAL NETWORK MODEL ADAPTATION FOR USE WITH

UNCONSTRAINED HANDWRITTEN TEXT RECOGNITION SYSTEM

E. Dolgova, D. Kurushin

*Perm State Technical University
29, Komsomolsky Avenue, Perm, Russia*

Abstract. The paper proposes an approach to adaptation of neural network models for creating OCR-systems, designed to work with unconstrained handwriting. The approach is based on the rejection of recognition of continuous characters and the transition to the recognition of individual strokes, which are then going to the characters and / or words of text. This approach can significantly reduce the dimension of neural networks used in the OCR-systems that will enhance their productivity and quality of recognition.

Key words: neural network, mathematical model, python, recognition, vector, count, language model.

004.932.1

ИССЛЕДОВАНИЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ ШТРИХОВ РУКОПИСНОГО ТЕКСТА

Е.В. Долгова, Д.С. Курушин

*Пермский государственный технический университет
614990, Пермь, Комсомольский пр., 29*

Аннотация: В статье предлагается и исследуется модель штрихов рукописного текста. Рассматриваемая модель представляет рукописный текст как совокупность штрихов нескольких известных классов. Модель позволяет абстрагироваться от начальных условий письма, таких как угол наклона, скорость, плотность и т.п. Благодаря процедуре векторизации для модели не имеет значения зашумленность исходного изображения и способ его получения. В работе проведен эксперимент, подтверждающий применимость модели для использования в задачах распознавания образов.

Ключевые слова: нейронная сеть, математическая модель, python, распознавание, вектор, граф, языковая модель, распознавание, штрих.

Широкое распространение и увеличение доступности технологий сканирования и цифрового фотографирования привело к быстрому росту цифровых коллекций документов. В таких коллекциях документы хранятся в виде растровых графических файлов. Оцифровка решает множество проблем, связанных с сохранением и организацией доступа к документам. Однако для реализации полнотекстового поиска, изучения содержания, подготовки публикаций требуется перевод из графического формата в текстовый, то есть распознавание текста.

Алгоритмы и программы автоматического распознавания текста разрабатываются уже несколько десятилетий. Можно сказать, что задача распознавания текстов на европейских языках, напечатанных на принтерах, решена.