Методы машинного обучения в задачах извлечения информации из текстов по эталону

© Алексеев С.С.
ООО «REHAU»
sergej.alexejew@rehau.com

© Морозов В.В.
ИАЦ Кортес
morozov@kortes.com

© Симаков К.В.
МГТУ им. Н.Э. Баумана
skv@ixlab.ru

Аннотация

Работа посвящена решению частного случая задачи извлечения информации из текстов – извлечению по эталону, при котором заранее известны эталонные (канонические) формы всех структур, подлежащих распознаванию в тексте. Основной акцент сделан на методах обучения, позволяющих снимать неоднозначности распознавания.

1 Введение

Задача извлечения информации из естественно-языковых текстов относится к классу задач распознавания. Извлечение заключается в выделении (распознавании) целевого текстового фрагмента, отвечающего определенным критериям, в сплошном тексте [7,24,27]. Задача извлечения информации возникает во многих областях, связанных с обработкой естественно-языковых текстов. Наиболее распространенным примером использования методов извлечения, является выделение в текстах участников событий заданного типа (например, событий, освещающих финансовые сделки между компаниями [10]).

Распознаваемые текстовые фрагменты являются результатом извлечения, а критерии, по которым они обнаруживаются – правилами извлечения. Составление полного перечня правил извлечения является весьма трудоемкой задачей независимо от формы их представления, поэтому параллельно с задачей извлечения решается задача автоматизированного составления правил распознавания [28]. Для этих целей обычно используют методы машинного обучения, позволяющие по набору обучающих позитивных и негативных примеров построить систему обобщенных правил.

2 Постановка задачи

2.1 Задача извлечения информации по эталону

Извлечение с использованием эталонной базы данных является частным случаем общей задачи извлечения информации из текстов. Несмотря на существование подходов к решению задачи в общем виде, извлечение информации по эталонной базе имеет свою специфику.

Положим, что в рамках некоторой предметной области существует эталонная база в виде реляционной таблицы C, содержащая N строк и M столбцов. В каждой строке c_i хранится некоторая структура данных, полям которой соответствуют столбцы таблицы. Если положить, что каждая строка c_i описывает некоторый объект, то таблица C в этом случае описывает класс однотипных сущностей предметной области.

В общем случае все сущности предметной области могут быть описаны в форме онтологии, где кроме классов будут установлены и отношения между ними, однако в данной работе интерес представляет только объекты одного класса, заключенные в таблице C.

Таким образом, каждая ячейка c_{ij} содержит значение j-ого свойства i-ого объекта. Положим, что каждое значение c_{ij} может быть приведено к текстовому представлению, содержащему одно и более слов естественного языка. Это представление является эталонным, поскольку отражает каноническую форму записи значений, принятую в данной предметной области. Например, канонической формой словосочетаний, образующих именную группу из существительного и прилагательного, может являться форма именительного падежа обоих слов данной группы. В более сложных многословных значениях c_{ij} каждое слово может употребляться в характерном для канонической формы падеже, роде, числе и т.д. Каноническая форма c_{ij} также определяет порядок следования слов в каждой записи.

Положим, что в рамках предметной области существуют тексты $T = \{t_k\}$, в которых в тех или иных контекстах могут принимать участие объекты таблицы C посредством употребления неканонических форм записи их свойств c_{ij}. Задача извлечения информации в данном случае заключается в установлении факта употребления конкретного объекта c_i в заданном тексте, сопровождающегося выделением

Труды 11[й] Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» - RCDL'2009, Петрозаводск, Россия, 2009.

текстовых фрагментов, содержащих неканонические формы записи его свойств c_{ij}.

Тот факт, что в текстах употребляются неканонические формы записи целевой информации, приводит к необходимости решать проблему неоднозначности. Задача снятия неоднозначности заключается в следующем. Из множества $\{c_i\}$, являющегося результатом распознавания для заданного текста t_k, необходимо выбрать единственный верный элемент c_r.

2.2 Примеры задачи извлечения по эталону

Наиболее характерной задачей, решение которой возможно с использованием методов извлечения по эталону, является построение графа цитирования. Суть задачи заключается в следующем. Имеется коллекция из N статей, каждая из которых характеризуется следующими свойствами: название, Ф.И.О. авторов, год публикации и наименование сборника трудов, в рамках которого статья опубликована.

Требуется проанализировать текст каждой статьи и выявить в каждом из них ссылки на другие статьи данной коллекции. Несмотря на то, что обычно ссылки устанавливаются в конце каждой статьи в разделе «Литература», а также на то, что существуют соответствующие правила их оформление, распознавание ссылок на практике является нетривиальной задачей. Проблемы возникают из-за орфографических ошибок, например, в написании Ф.И.О. или в названии статьи, из-за неполноты приводимой информации, а также из-за нарушения порядка слов в некоторых полях библиографической ссылки.

Другим примером является задача распознавания в текстах почтовых адресов. Предположим, имеется N почтовых адресов, каждый из которых представлен набором полей (регион, район, город, улица и т.д.). Также предположим, что имеется поток документов, которые необходимо раскладывать по группам в зависимости от адреса, указанного в тексте документа. Форма употребления адресов в текстах может иметь произвольный характер, существенно отличающийся от канонической записи. Например, в адресе не всегда указывается регион и район, порядок слов в многословных наименованиях может отличаться от порядка их следования в канонической форме, некоторые слова в названиях могут записываться в тексте в сокращенной форме, либо с орфографическими ошибками (что вообще характерно для любых имен собственных, не подчиняющихся общей грамматике языка). В такой постановке, задача выявления адреса в тексте также становится нетривиальной, несмотря на то, что в наличии имеется эталонная база адресов.

3 Обзор методов извлечения

В настоящий момент в литературе не существует отдельных упоминаний о рассматриваемом подклассе задач, поскольку многие исследователи решают задачу извлечения информации из текстов в общем виде. В качестве основных направлений в данной области можно выделить символьный и численный подходы.

3.1 Символьный подход

В рамках данного подхода правила извлечения записываются на формальном языке, напоминающем язык регулярных выражений. Такие языки позволяют описывать свойства контекстов употребления целевой информации в виде морфологических признаков слов, окружающих извлекаемые текстовые фрагменты, синтаксические роли слов и групп слов, а также конкретные ключевые слова-сателлиты, регулярно встречающиеся в контексте целевой информации. Данные методы разделяются на два класса: пропозиционные [4,11,15,17] и реляционные [3,5,6,16] - в зависимости от выразительных возможностей языка и возможностей метода обучения.

В пропозиционных методах язык правил извлечения эквивалентен логике нулевого порядка. Полагается, что шаблоны правил, определяющие ограничения на связи между словами, задаются экспертом, а метод обучения автоматически подбирает ограничения, накладываемые на значения свойств слов-заполнителей этих шаблонов. В реляционных методах язык правил извлечения эквивалентен логике первого порядка [19], а методы обучения автоматически формируют как ограничения на связи между словами в тексте, так и ограничения на слова, принимающие участие в этих связях.

Обучение в символьных методах организуется по принципу дедуктивного вывода, либо по принципу индуктивного обобщения. В обоих случаях эксперт готовит обучающие примеры, представляющие собой тексты, в которых явным образом выделены целевые фрагменты, подлежащие извлечению. Далее к обучающей выборке применяется выбранный метод обучения, синтезирующий обобщенные правила распознавания целевой информации. Полученные таким образом правила, по сути, заключают в себе закономерности, характерные для употребления целевой информации в текстах обучающей выборки.

3.2 Численный подход

Методы данного класса полагают, что изначально имеется набор элементарных правил, определенный априори, а задача построения извлекателя сводится к подбору композиции этих правил, обеспечивающей заданную точность и полноту извлечения на исходной обучающей выборке. Композиция в разных методах может строиться по-разному, но общим у этих методов является то, что каждое элементарное правило включается в композицию с определенным численным весом.

Из наиболее распространенных представителей данного класса следует выделить Байесов классификатор [14], Скрытые Марковские Модели [2],

методы максимизирующие энтропию [1,9] и условные случайные поля [12]. Обучение во всех перечисленных методах сводится к подбору вероятностных коэффициентов, оценивающих вклад элементарного правила распознавания в рамках их полной композиции.

3.3 Применимость к поставленной задаче

Символьные методы естественным образом допускают использование ключевых слов в правилах извлечения, так что в общем случае правило извлечения может содержать прямое перечисление всех слов эталонной базы, однако при достаточно больших N (например, для базы почтовых адресов $N\sim10^6$) такое использование правил будет неэффективным. Вместе с тем, такое перечисление не дает ответ на вопрос о том, как распознавать в текстах неканонические формы перечисленных слов. Проблема неоднозначного распознавания в методах данного класса также не решается.

Численные методы позволяют решить задачу снятия неоднозначности распознавания, поскольку каждому из вариантов извлечения может быть поставлено в соответствие число в виде вероятности распознавания, так что выбор единственного варианта становится очевидным.

Общим свойством указанных подходов является то, что они запоминают структуру целевой информации и стремятся не запоминать конкретные ключевые слова контекста извлекаемых данных, чтобы добиться должного уровня общности синтезируемых правил. Однако в нашей задаче ситуация полностью противоположная – за счет того, что имеется полная эталонная база, появляется возможность не запоминать структуру извлекаемой информации. Поэтому порядок следования слов и распознаваемых полей в тексте становится неважным. Возвращаясь к описанным выше примерам, автор статьи может идти как перед ее названием, так и после нее, аналогично в записи почтового адреса все его поля могут быть записаны в любом порядке.

4 Метод извлечения

Наличие эталонной базы позволяет реализовать достаточно простой способ выделения целевой информации на основе технологии полнотекстового поиска.

Упрощенно данный метод может быть описан следующим образом. Для таблицы C строится полнотекстовый индекс, представляющий собой словарь, в котором объединены все слова, задействованные в ячейках c_{ij}. Для каждого слова формируется инвертированный список, содержащий номера строк c_i, в полях которых встречается данное слово.

При анализе по обрабатываемому тексту скользит окно, размер которого соответствует длине максимальной строки c_{max} исходной таблицы. На основе каждого слова окна формируется поисковый запрос, в котором логическим ИЛИ объединены все его возможные словоформы, которые кроме морфологического словоизменения могут учитывать вероятные опечатки и варианты сокращения. Данный запрос выполняется на имеющемся словаре. Результатом такого поиска является список строк исходной таблицы, в каждой из которых встречается один или несколько вариантов словоизменения исходного слова. Дополнительному учету могут подлежать слова, по которым ничего не удалось найти. После превышения заданного порога по числу таких слов появляется возможность сделать вывод о том, что текущий текстовый сегмент не содержит целевой информации, после чего сдвинуть окно дальше, пропустив этот сегмент.

Найденные по всем словам окна списки пересекаются, в результате чего получается итоговый список $C_t = \{c_i\}$, содержащий номера записей исходной таблицы, в которых встречается большинство слов текущего окна. Таким образом, встает вторая задача – выбор одного единственно верного варианта распознавания c_r из полученного множества C_t.

Для выполнения такого выбора необходима некоторая функция $\rho : T \times C \to [0..1]$, позволяющая для каждого варианта распознавания (t, c_i) (где t – распознанный фрагмент текста, c_i - распознанный объект) поставить в соответствие число, оценивающее качество распознавания данного варианта. Имея такую функцию, выбор единственно наилучшего варианта распознавания выполняется согласно критерию максимизации значения ρ, т.е. $(t, c_r) = \arg\max_{c \in C_t} \rho(t, c)$.

5 Методы обучения извлекателя

Построение функции $\rho : T \times C \to [0..1]$ посвящена вторая часть работы. Чтобы получить функцию данного вида, необходимо располагать числовыми свойствами варианта распознавания. Пусть введено семейство признаков $\{f_j\}_{j=1}^{F}$, каждый признак представляет собой функцию вида $f_j : T \times C \to R$, т.е. позволяет дать количественную оценку варианту распознавания $(t, c) \in T \times C$. Тогда искомую функцию ρ можно искать в виде композиции $\rho(t, c) = (f_1 \circ f_2 \circ ... \circ f_F)(t, c)$, где символ $\circ$ обозначает некоторую композицию функций.

Признаки $\{f_j\}_{j=1}^{F}$ могут описывать различные свойства варианта распознавания c_i такие, как позиционную близость распознанных полей c_{ij}, количество распознанных полей $|c_{ij}|$, количество слов с опечатками, общее количество слов и др.

Композицию $(f_1 \circ f_2 \circ ... \circ f_F)(t, c)$ можно сконструировать, обладая набором обучающих примеров $T_{teach} = \{t_i\}$. Где каждый позитивный пример пред-

ставляет собой пару (t_i, c_p^i) так, что для каждого текстового фрагмента $t_i \in T_{teach}$ явно выделен правильный вариант распознавания c_p^i. Негативными обучающими примерами для каждого t_i объявляется множество $\{c_n^i\} = C_i \setminus c_p^i$, где C_i - все варианты распознавания в рамках фрагмента t_i.

Положим, что $\rho(t_i, c_p^i) = 1$ и $\rho(t_i, c_n^j) = 0 : c_n^j \in C_i \setminus c_r^i$. Тогда обучающую выборку можно представить в виде таблицы следующего вида.

Табл. 1. Обучающая выборка для синтеза $\rho(t,c)$

$\rho(t,c)$	$f_1(t,c)$	$f_2(t,c)$	…	$f_F(t,c)$
…	…	…	…	…
$\rho(t_i, c_p^i) = 1$	$f_1(t_i, c_p^i)$	$f_2(t_i, c_p^i)$	…	$f_F(t_i, c_p^i)$
$\rho(t_i, c_n^1) = 0$	$f_1(t_i, c_n^1)$	$f_2(t_i, c_n^1)$	…	$f_F(t_i, c_n^1)$
…	…	…	…	…
$\rho(t_i, c_n^j) = 0$	$f_1(t_i, c_n^j)$	$f_2(t_i, c_n^j)$	…	$f_F(t_i, c_n^j)$
…	…	…	…	…

Эту таблицу далее будем назвать интерполяционной, поскольку она содержит значения целевой функции $\rho(t,c)$ в интерполяционных точках F-мерного пространства, где каждой j-ой координате соответствует признак f_j вариантов распознавания. Далее обозначим число интерполяционных точек как N_t.

Каждый вариант распознавания характеризуется F-мерным вектором значений числовых признаков $\{f_j\}_{j=1}^F$, обозначив его как x, можем перейти от представления функции $\rho(t,c)$ к представлению $\rho(x)$. Через x_j будем обозначать j-ую координату вектора, т.е. $x_j \equiv f_j(c,t)$.

Таким образом, задача обучения извлекателя сводится к задаче аппроксимации функции $\rho(x)$ по заданной интерполяционной таблице, где в качестве аргумента выступают F-мерные точки (векторы).

5.1 Наивный Байесов классификатор

Данный способ аппроксимации $\rho(x)$ был выбран в качестве отправной точки, относительно которой выполнялось сравнение остальных методов обучения в контексте поставленной задачи. Поскольку целевая функция в идеале может принимать только два значения 1 и 0, с точки зрения классификатора, они расцениваются как два класса.

Классификатор работает на основе формулы $p(\rho \mid x_1 \cdot \ldots \cdot x_F) = \dfrac{p(\rho) \cdot \prod_{j=1..F} p(x_j \mid \rho)}{\prod_{j=1..F} p(x_j)}$, где $p(\rho)$, $p(x_i \mid \rho)$ и $p(x_i)$ - распределения вероятностей, формируемые в результате обучения. Формула справедлива, при условии, что признаки $\{x_j\}_{j=1}^F$ независимы.

Основная проблема такого подхода заключается в том, что признаки могут принимать бесконечно большое число значений, тогда как на обучающей выборке может быть получено распределение только для тех значений, которые попали в выборку. Для решения этой задачи значения всех признаков были приведены к диапазону [0…1]. Этот диапазон был разделен на 10 частей так, что разные значения любого j-ого признака, попадающие в один и тот же диапазон, рассматривались в качестве одного и того же значения случайной величины x_j. Таким образом, распределение $p(x_i)$ аппроксимируется ступенчатой функцией вида

$$p(x_i) = \begin{bmatrix} p(0 \le x_i < 0.1) \\ \ldots \\ p(0.9 \le x_i \le 1) \end{bmatrix}.$$

Вероятности в правой части выражения определяются на обучающей выборке путем подсчета соответствующих относительных частот. Аналогичным образом формируются распределения $p(x_i \mid \rho)$.

Из достоинств данного подхода можно выделить простоту реализации. К недостаткам отнесем необходимость грубой аппроксимации $p(x_i)$ и $p(x_i \mid \rho)$ в виде ступенчатых функций, а также наличие условие независимости признаков $\{x_j\}_{j=1}^F$. В нашем случае за каждым признаком x_j, по сути, стоит функция $f_j(c,t)$, реализуемая алгоритмически, что фактически не дает информации о наличии/отсутствии зависимостей между ними.

5.2 SVM классификатор

Идея использования данного метода возникла из предположения о том, что два множества интерполяционных точек со значениями $\rho(x) = 1$ и $\rho(x) = 0$ в идеале являются выпуклыми непересекающимися множествами [23]. В этом случае, можно попытаться найти разделяющую их гиперплоскость $\langle w, x \rangle = w_0$, равноудаленную от границ этих множеств, а затем использовать ее в работе распознавателя (здесь w - нормаль к гиперплоскости, w_0 - число, задающее ее сдвиг). Целевая функция $\rho(x)$ примет в этом случае вид

$$\rho(x) = \begin{bmatrix} 1, & \textit{если} \quad \langle w, x \rangle \ge w_0 \\ 0, & \textit{если} \quad \langle w, x \rangle < w_0 \end{bmatrix}.$$

Метод опорных векторов (SVM) позволяет найти разделяющую гиперплоскость, даже если исходные множества линейно неразделимы, в этом случае итоговый классификатор будет работать с некоторой ошибкой. Для апробации этого метода интерполяционная таблица использовалась как есть без дополнительных преобразований. В качестве SVM реализации использовался проект SVM-Light [18].

5.3 МНК аппроксимация

Одним из способов синтеза $\rho(x)$ является ее представление в виде линейного разложения в функциональном базисе, т.е. $\rho(x)=\sum_{j=1}^{M_B} K_j \cdot \beta_j(x)$, где β_j - j-ая базисная функция, M_B - число базисных функций, $K_j \in R$ - коэффициенты разложения.

Для нахождения коэффициентов K_j методом наименьших квадратов [22] отыскивается минимум суммы квадратов отклонений вида $E_1=\sum_{i=1}^{N_t}\left(\rho^i-\sum_{j=1}^{M_B} K_j\cdot\beta_j(x^i)\right)^2$, где x^i - i-ая интерполяционная точка, а ρ^i - значение целевой функции в этой точке. Для этого необходимо решить систему из M_B линейных уравнений вида $\frac{\partial E_1}{\partial K_j}=0$. Итоговый вид j-ого уравнения следующий $\sum_{k=1}^{M_B} A_{kj}\cdot K_k = B_j$, где $A_{kj}=\sum_{i=1}^{N_t}\beta_k(x^i)\cdot\beta_j(x^i)$ и $B_j=\sum_{i=1}^{N_t}\rho^i\cdot\beta_j(x^i)\cdot$

В экспериментах в качестве базиса использовались полиномы следующего вида:

$$\begin{cases}\beta_k=1, & если \quad k=1\\ \beta_k=(x_j)^n, & если \quad k=F\cdot(j-1)+n+1,\ j=1..F, n=1..D_n\end{cases}$$

где D_n варьировалась от 2 до 9. Также вместо обычных полиномов использовались полиномы Чебышева до 3 степени.

Основным недостатком этого подхода является отсутствие гарантий того, что после определения K_j будет иметь место $\sum_{j=1}^{M_B} K_j\cdot\beta_j(x^i)=\rho^i$, а также отсутствие способов, позволяющих как-то влиять на итоговый результат, за исключением возможности выбрать сам функциональный базис. На практике это означает, что обученный извлекатель будет неправильно выбирать варианта распознавания c_r из множества C_t для обучающего примера $t_i \in T_{teach}$. Эта проблему далее будем назвать проблемой недостаточной обученности.

5.4 МГУА аппроксимация

Одним из подходов к решению проблемы недостаточной обученности является наращивание степени D_n полиномов, в теории при $D_n = N_t$ МНК гарантирует совпадение синтезированной функции с ожидаемыми значениями в интерполяционных точках, однако при $N_t \in [10^2...10^3]$ реализация такого подхода становится непрактичной.

Отчасти данную проблему можно решить, используя метод группового учета аргументов (МГУА) [8]. В данной работе был реализован МГУА с линейными частными описаниями вида $\rho_{kj}(x)=K_0+K_k\cdot x_k+K_j\cdot x_j$. В качестве критерия регулярности в экспериментах использовалась выражение вида

$$\delta^2=\frac{1}{N_t}\left(\sum_{i=1}^{N_t}\left(\rho_{kj}(x^i)-\rho^i\right)^2+\sum_{i=1}^{N_t} err(x^i)\right),$$

где $err(x^i)=1$, если на этапе селекции на тестовом наборе частное описание ρ_{kj} делает неправильный выбор среди C_t вариантов распознавания на тексте t, одному из которых соответствует точка x^i, в противном случае $err(x^i)=0$. Коэффициенты K_k и K_j на каждой итерации определяются методом МНК.

Достоинством данного подхода является возможность на каждой итерации учитывать семантику значений ρ^i. Так в нашем случае при выборе предпочтительных ρ_{kj} из текущего ряда селекции, кроме среднеквадратичной ошибки учитывается привязка конкретных точек к вариантам распознавания одного и того же обучающего примера t. Это позволяет выбирать частные описания ρ_{kj}, которые не только в среднем мало ошибаются согласно δ^2, но к тому же позволяют принимать корректные решения на максимальном числе обучающих примеров.

Недостатком данного подхода является отсутствие гарантий, что при наличии наложенных ограничений процесс обучения сойдется за конечное число итераций, а также то, что на некоторых итерациях могут быть ошибочно отброшены значимые переменные.

5.5 Деревья решений

Деревья решений, обычно, используются для классификации при помощи правил в иерархической, последовательной структуре, данное свойство позволяет использовать деревья решений для аппроксимации функции, имеющей конечное число дискретных значений, такой как $\rho(x)$ [20].

Известен ряд алгоритмов для построения дерева решений, таких как CART, ID3, C4.5 и некоторые другие [13,21], вне зависимости от деталей реализации, данные алгоритмы разбивают множество числовых признаков $X=\{x\}$ на подмножества, каждой из которых ассоциировано с одним из значений функции $\rho(x)$. Очевидно, что можно ожидать удовлетворительного результата только в случае, если количество таких подмножеств конечно и «хорошо» охватывается обучающей выборкой. Отдельно от-

метим, что из-за обрезки деревья решений не гарантируют прохождение синтезированной функции через все интерполирующие точки $\{\rho^i\}$.

Вместе с тем, деревья решений могут обеспечить высокую точность распознания лишь на линейно разделимых множествах. Так как рассматриваемая в данной работе задаче не гарантирует линейной разделимости множеств $\{x:\rho(x)=1\}$ и $\{x:\rho(x)=0\}$, то представляет особый интерес экспериментальная апробация данного метода.

5.6 Нейронные сети

Для решения задач аппроксимации наиболее подходящими являются многослойные сети прямого распространения - многослойные персептроны и нейронные сети, использующие радиальные базисные функции [25]. Помимо входного и выходного слоя, нейронная сеть может содержать один или более скрытых слоев, количество которых выбирается на основе эмпирических критериев. Отметим, что использование персептрона без скрытых слоев возможно только в случае линейно разделимых множеств.

Для данной задачи была выбрана модель персептрона с одним скрытым слоем из 9 нейронов и 9 входных нейронов, что должно позволить данной модели обеспечивать точность на уровне модели МНК. Так как данные для обучения являются непрерывными, то в качестве функции активации использовалась сигмоидальная функция.

Для обучения многослойной сети обычно используется алгоритм обратного распространения ошибки. При этом в качестве критерия остановки обучения можно использовать критерий полного распознания всех примеров обучающего множества $\{\rho^i\}$, т.к. исходя из теоремы сходимости персептрона, можно обеспечить прохождение синтезируемой функции через все интерполирующие точки [26], что, однако, может потребовать неприемлемого большого времени обучения.

6 Экспериментальное сравнение методов обучения

6.1 Описание эксперимента

В рамках данной работы были проведены эксперименты, направленные на сравнение и анализ рассмотренных методов обучения на предмет их практической применимости в задаче извлечения почтовых адресов России в произвольных текстах в рамках on-line сервиса «Охотник за адресами» (http://www.ahunter.ru). За основу эталонной базы почтовых адресов был взят классификатор КЛАДР, для которого был построен полнотекстовый индекс в соответствии с положениями, изложенными выше.

В качестве признаков $\{f_j\}_{j=1}^{F}$ были выбраны:

- $f_1(t,c)$ - инвертированная сумма расстояний Левенштейна по всем распознанным адресным полям c_{ij} и соответствующим им текстовым написаниям в t (чем ближе каноническое написание c_{ij} к его неканоническому представлению в t, тем $f_1(t,c)$ больше);
- $f_2(t,c)$ - количество слов в текстовом фрагменте t, не задействованных при распознавании c;
- $f_3(t,c)$ - количество распознанных числовых полей адреса (номер дома и пр.);
- $f_4(t,c)$ - количество верифицированных числовых полей адреса (номер дома и пр.);
- $f_5(t,c)$ - количество распознанных полей, содержащих тип адресного объекта (город, улица, бульвар, проспект и пр.);
- $f_6(t,c)$ - суммарное число слов во фрагменте t, задействованных при распознавании c;
- $f_7(t,c)$ - относительная позиция первого распознанного слова внутри фрагмента t;
- $f_8(t,c)$ - относительная позиция последнего распознанного слова текстового фрагмента t;
- $f_9(t,c)$ - количество полей в c с устаревшими или синонимичными названиями.

Обучающая выборка была построена на основе анализа журналов работы сервиса по следующему принципу. Было отобрано ~320 текстовых фрагментов, на которых пользователи сервиса получили неоднозначные результаты. Эти тексты были вручную размечены, так, что в них посредством специальных тэгов явным образом были выделены поля извлекаемой адресной структуры. Пример такого текста приведен в таблице 2.

Табл. 2. Пример размеченного текста.

Текст	Москва 2-ая Бауманская 5
Разметка	<A:Region>Москва</A:Region> <A:Street>2-ая Бауманская</A:Street> <A:House>5</A:House>

Каждый неразмеченный текст t подавался на вход извлекателю в режиме обучения, при котором он выдавал все варианты извлечения C_t. Для каждого варианта $c_i \in C_t$ рассчитывался вектор значений признаков $\{f_j(c_i,t)\}_{j=1}^{9}$. Далее выполнялась покоординатная нормировка: значения каждого признака $f_j(c_i,t)$ у всех вариантов $c_i \in C_t$ делились на $\max_{c_i \in C_t} f_j(t,c_i)$. Вариант извлечения $c_p \in C_t$, совпадающий с показаниями разметки объявлялся позитивным и для него принималось значение $\rho(t,c_p)=1$, для остальных вариантов $c_n \in C_t \setminus c_p$ принималось $\rho(t,c_n)=0$.

Таким образом, была заполнена интерполяционная таблица (см. табл. 1). После устранения дублей, возникающих в результате покоординатной нормировки, в таблице было оставлено 520 интерполяци-

онных точек, на которых и проводились эксперименты.

В рамках эксперимента были подготовлены срезы полной обучающей выборки, содержащие 5, 10, 15, 20, 25, 30, 35, 40, 50, 60, 70, 80, 90 и 100 процентов ее интерполяционных точек. Каждый метод обучения, таким образом, применялся к каждому из указанных срезов.

Тестирование каждой обученной модели проводилось на полной выборке в 520 интерполяционных точек. При этом подсчитывалось число ложных срабатываний, которые использовались для расчета точности обученной модели согласно выражению $P = \frac{N_s - E}{N_s}$, где N_s - число исходных текстовых примеров (не интерполяционных точек) по отношению к которым применяется тестируемый метод снятия неоднозначности распознавания, E - число текстовых примеров, на которых распознаватель допустил ошибку.

6.2 Результаты эксперимента

В таблице 3 приведены значения точности распознавания в зависимости от метода обучения и от размера обучающей выборки.

В первой колонке таблицы 3 приведены названия методов, принимавших участие в экспериментах для обучения распознавателя. Остальные колонки соответствуют размерам выборки, на которой проводилось обучение. Числа в прочих ячейках таблицы отражают значения точности распознавателя, обученного по методу, соответствующему строке ячейки, и на выборке, соответствующей колонке ячейки.

Как отмечалось, в обучении по МНК пробовались разные функциональные базисы (полиномы различных степеней и полиномы Чебышева), которые, как показали эксперименты, не влияют коренным образом на характер зависимости точности от размера выборки. Поэтому предпочтение было отдано базису $\beta_j(x) = x_j$, что фактически соответствует представлению $\rho(x) = \langle K \cdot x \rangle$, где K - вектор коэффициентов разложения. В таблице 3 этому эксперименту соответствует строка с названием «МНК (линейный)». Также в таблице 3 приведены результаты экспериментов для МНК с разложением по базису полиномов Чебышева до 3-ей степени, этим результатам соответствует строка с названием «МНК (Чебышев)». На рис. 1 отражены диаграммы, демонстрирующие небольшую разницу между этими двумя видами МНК.

На рис. 2 приведено графическое представление полученных данных по всем методам из табл. 3, кроме МНК с разложением по базису полиномов Чебышева.

В случае с линейным МНК, использующим базис $\rho(x) = \langle K \cdot x \rangle$, представляет интерес его сравнение с SVM, поскольку они реализуют одну и ту же идею разделяющей гиперплоскости, но различными способами. Полученные показатели точности указывают на небольшое превосходство SVM, которое, однако, было достигнуто за счет опытного подбора управляющего параметра $C \geq 100$ (масштабный коэффициент, позволяющий задать компромисс между шириной зазора между разделяемыми множествами и суммарной ошибкой классификации по всем обучающим примерам [18]).

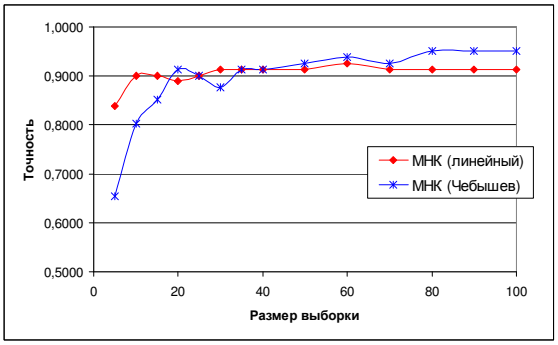

Рис. 1. Сравнение двух видов МНК.

При выполнении аппроксимации МГУА сходимость процесса обучения ограничивалась скоростью обучения так, что процедура завершалась, когда минимальное значение критерия регулярности на текущем слое селекции уменьшалось менее чем на 1% по отношению к этому же значению от предыдущего слоя. На каждом слое селекции оставлялось до 9 узлов с частными описаниями. Результирующие структуры, формируемые по данному методу, при обучении на разных выборках включали от 4 до 6 слоев.

Примечательным является тот факт, что для большинства методов (все кроме дерева решений и персептрона) рост размера выборки не гарантирует возрастания точности распознавателя. Например, для линейного МНК и SVM имеет место убывание точности между точками 15% и 20%.

У методов МНК, SVM, МГУА и классификатора Байеса общим свойством является недостаточная обученность, фактически не позволяющая гарантировать отсутствие ошибок распознавания на тех же данных, на которых обучался распознаватель.

Вместе с тем эти методы демонстрируют быструю сходимость процесса обучения. Так что, начиная с обучающей выборки, содержащей 35% примеров от тестового множества, возрастание точности распознавания практически прекращается. Данный факт говорит о хорошей способности к обобщению входных данных у этих методов обучения.

Перечисленные три свойства: быстрая сходимость, колебания точности и недостаточная обученность - можно объяснить ограниченными выразительными возможностями моделей, лежащих в основе этих методов. Фактически для этих методов характерно жесткое определение структуры обучаемой модели: для МНК и МГУА – линейная комбинация базисных функций, для SVM – уравнение гиперплоскости, для классификатора Байеса – ступенчатые функции распределения вероятностей.

Таблица 3. Точность обученных распознавателей на разных выборках.

	5%	10%	15%	20%	25%	30%	35%	40%	50%	60%	70%	80%	90%	100%
МНК (линейный)	0,8395	0,9012	0,9012	0,8889	0,9012	0,9136	0,9136	0,9136	0,9136	0,9259	0,9136	0,9136	0,9136	0,9136
МНК (Чебышев)	0,6543	0,8025	0,8519	0,9136	0,9012	0,8765	0,9136	0,9136	0,9259	0,9383	0,9259	0,9506	0,9506	0,9506
МГУА	0,8395	0,8889	0,9259	0,9259	0,9383	0,9383	0,9259	0,9383	0,9259	0,9259	0,9383	0,9630	0,9506	0,9753
Байес	0,3210	0,6914	0,7407	0,7531	0,7901	0,7778	0,7778	0,7654	0,7654	0,7778	0,7901	0,8025	0,8025	0,8025
SVM	0,9012	0,9506	0,9383	0,9136	0,9259	0,9383	0,9259	0,9259	0,9259	0,9259	0,9506	0,9506	0,9383	0,9383
Дерево решений	0,2963	0,6049	0,6296	0,6296	0,6420	0,6420	0,6420	0,6420	0,6543	0,6667	0,6667	0,6914	0,7531	0,7778
Персептрон	0,4321	0,4938	0,5432	0,6049	0,6914	0,7284	0,7407	0,7654	0,7778	0,8025	0,8148	0,8765	0,9259	0,9506

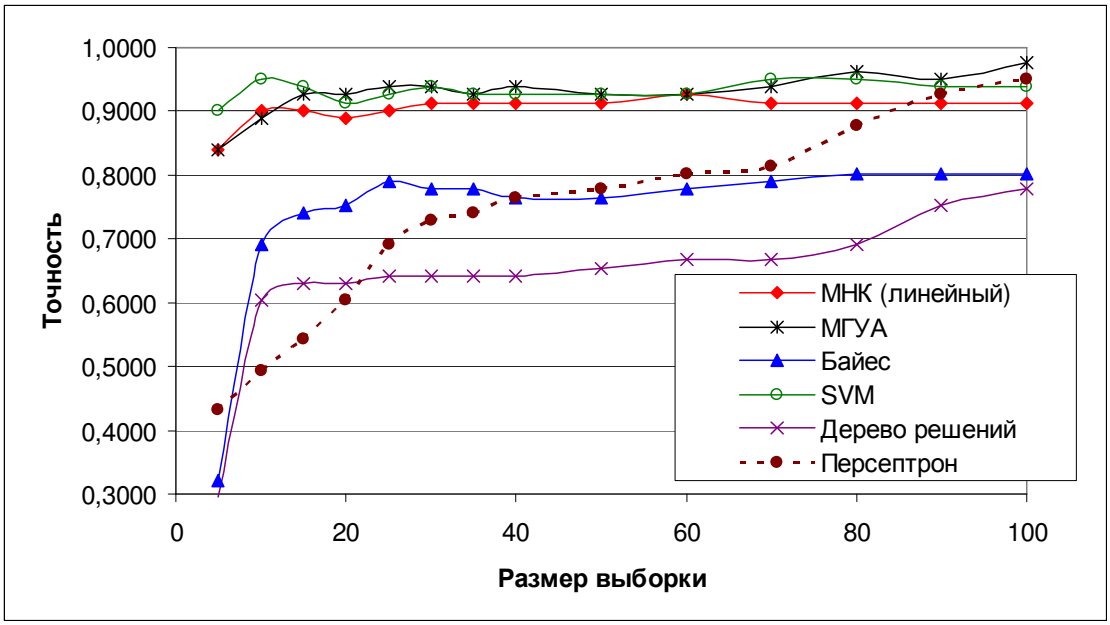

Рис. 2. Точность обученных распознавателей.

Информации в небольшой обучающей выборке оказывается достаточно, чтобы настроить все параметры данных моделей и достичь предела их выразительных возможностей. С дальнейшим ростом обучающей выборки возрастает количество информации, которое нужно учесть в параметрах модели, что фактически не приводит к качественному росту точности, а вызывает лишь небольшие колебания относительно достигнутого порога.

Так же следует учитывать, что с ростом объема обучающей выборки возрастает и ее зашумленность, обусловленная ошибками экспертов при подготовке выборки.

Деревья решений, как и ожидалось, показали наихудший результат, что связанно как с простотой данной модели, так и с тем, что к построенному дереву применялась процедура подрезки тех ветвей, уровень доверия к которым был менее 80%. Исключение процедуры подрезки ветвей могло бы несколько улучшить результат, но, вместе с тем, на больших массивах данных это могло бы привести к переобученности модели. Хорошо заметно, что точность распознания данного метода сравнима с классификатором Байеса, что связанно с тем, что в основе данных моделей лежит схожий вероятностный принцип выбора решения.

Нейронные сети, а именно персептрон с одним скрытым слоем, показал хорошую точность распознания в случаях, когда обучающее множество покрывало более 60% тестового множества. Плохие результаты при обучающем множестве менее 30% от тестового объяснимы тем, что для данной модели персептрона — 9 входных нейронов и один скрытый слой, содержащий 9 нейронов, эти выборки недостаточны для обучения и вызывают явление, известное как переобученность.

Проведенные эксперименты для модели персептрона без скрытых слоев показали, что данная модель даже на полной выборке обеспечивает невысокую точность, что связанно с линейной неразделимостью обучающего множества. Численные результаты по этим экспериментам в работе не приводятся.

6.3 Выводы по экспериментам

Для данной задачи была также выполнена ручная аппроксимация, которая заключалась в подборе экспертом эвристики, описывающей, с его точки зрения, функцию $\rho(x)$ наилучшим образом. Точность этой эвристики на тестовом наборе составила 0.93, поэтому целесообразность использования рассмотренных методов обучения оценивалась на основе сравнения этого значения с их показаниями точности.

Деревья решений, классификатор Байеса и однослойный персептрон показали свою неприменимость к решению исходной задачи. Несмотря на то, что данные модели обладают высокой скоростью работы, их точность оказалась ниже точности эвристической аппроксимации.

Методы МНК и SVM обеспечивают приемлемую точность распознавания, однако для них не существует способов управления процессом обучения, позволяющих влиять на итоговую точность. Метод МГУА показал лучший результат, причиной чего стала возможность управления процессом обучения, путем введения на этапе селекции релевантного для поставленной задачи критерия регулярности.

Многослойный персептрон продемонстрировал высокую точность распознания при представительной обучающей выборке (сравнимой с адаптированным к задаче методом МГУА) однако на данной выборке не удалось оценить обобщающие возможности этой модели.

7 Заключение

В данной работе предложен метод извлечения целевой информации по эталону. Полагается, что система извлечения обладает полной априорной базой так, что распознавание в текстах возможно только той информации, которая имеется в этой базе. Не смотря на кажущуюся ограниченность такого подхода, на практике он находит применение в различных областях таких, как распознавание Ф.И.О. и извлечение топонимов в произвольных текстах. Отсутствие сильной зависимости от формы текста является главным достоинством такого подхода, вытекающим из того, что не возникает необходимости запоминать структуру текста, в отличие от методов извлечения общего вида.

Авторами также разработан метод снятия неоднозначности распознавания, являющейся ключевой проблемой в задачах извлечения информации из текстов. Решение этой задачи рассмотрено с позиции машинного обучения на примерах. Проанализировано несколько вариантов такой реализации, каждая из которых проверена экспериментально на задаче выявления почтовых адресов России в произвольных текстах. Среди рассмотренных способов обучения наиболее предпочтительными являются те из них, которые оперируют многослойными структурами. В нашем случае к ним относится многослойный персептрон и МГУА-аппроксиматор.

Разработанные методы реализованы в виде прототипа системы извлечения по эталону, работоспособность которого можно проверить on-line по адресу: http://www.ahunter.ru.

Литература

[1] Berger A.L., Della Pietra V.J., Della Pietra S.A. A maximum entropy approach to natural language processing // Computational Linguistics archive. – 1996. – Vol. 22, Issue 1, – P. 39–71.

[2] Borkar V., Deshmukh K., Sarawagi S. Automatic segmentation of text into structured records // Proceedings of the 2001 ACM SIGMOD international conference on Management of data. – 2001. – P. 175–186.

[3] Califf M.E., Mooney R.J. Bottom–up relational learning of pattern matching rules for information extraction // Journal of Machine Learning Research. – 2003. – Vol. 4. – P. 177–210.

[4] Chai J.Y., Biermann A.W., Guinn C.I. Two dimensional generalization in information extraction // In Proceedings of the Sixteenth National Conference on Articial Intelligence. – 1999. – July. – P. 431–438.

[5] Dejean H. Learning rules and their exceptions // The Journal of Machine Learning Research archive. – 2002. – Vol. 2 (March). – P. 669–693.

[6] Freitag D. Machine Learning for Information Extraction in Informal Domains // Machine Learning. – 2000. – Vol. 7. – P. 169–202.

[7] Grishman R., Sundheim B. Message Understanding Conference–6: a brief history // Proceedings of the 16th conference on Computational linguistics. – 1996. – Vol.1. – P. 466 – 471.

[8] Group Method of Data Handling [Электронный ресурс] – Режим доступа: http://www.gmdh.net/, свободный.

[9] Hai Leong Chieu, Hwee Tou Ng. A maximum entropy approach to information extraction from semi–structured and free text // Eighteenth national conference on Artificial intelligence. – 2002. – P. 786–791.

[10] Huffman S.B. Learning to extract information from text based on user–provided examples // Proceedings of the fifth international conference on Information and knowledge management. – Rockville, Maryland, (United States), 1996. – P. 154–163.

[11] Kim J., Moldovan D. Acquisition of Linguistic Patterns for Knowledge–Based Information Extraction // IEEE Transactions on Knowledge and Data Engineering archive. – 1995. – Vol. 7, Issue 5. – P. 713–724.

[12] McCallum A. An Introduction to Conditional Random Fields for Relational Learning / C. Sutton, A. McCallum // Introduction to Statistical Relational Learning / Edited by Lise Getoor and Ben Taskar. – MIT Press, 2007. – P. 95–130.

[13] Murthy S. Automatic construction of decision trees from data: A Multi-disciplinary survey 1997
[14] Pedersen T. A simple approach to building ensembles of Naive Bayesian classifiers for word sense disambiguation // Proceedings of the first conference on North American chapter of the Association for Computational Linguistics. – 2000. – P. 63 – 69.
[15] Riloff E. Automatically Constructing a Dictionary for Information Extraction Tasks // In Proceedings of the 11th National Conference on Artificial Intelligence (AAAI). – 1993. – P. 811–816.
[16] Soderland S. Learning information extraction rules for semi–structured and free text. Machine Learning. – 1999. – Vol. 34, Issue 1–3. P. 233–272.
[17] Soderland S. Crystal: Inducing a conceptual dictionary / S. Soderland, D. Fisher, J. Aseltine, We. Lehnert // In Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence. – 1995. – P. 1314–1319.
[18] SVM-Light Support Vector Machine [Электронный ресурс] – Режим доступа: http://svmlight.joachims.org/, свободный.
[19] Turmo J., Ageno A., Catala N. Adaptive information extraction // ACM Computing Surveys archive. – 2006. – Vol. 38, Issue 2. – Article No. 4.
[20] Venkatesan T. Chakaravarthy, Vinayaka Pandit, etc. Decision trees for entity identification: approximation algorithms and hardness results. //Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, pp 53 – 62, 2007.
[21] Yuan Y. and Shaw M. J. Induction of fuzzy decision trees - Fuzzy Sets Syst., vol. 69, pp. 125-139, 1995.
[22] Аппроксимация методом наименьших квадратов (МНК) [Электронный ресурс] – Режим доступа: http://alglib.sources.ru/interpolation/linearleastsquares.php, свободный.
[23] Карманов В.Г. Математическое программирование: учебное пособие. – 5-ое изд., стереотип. – М.: ФИЗМАТЛИТ, 2004. – 264 с.
[24] Кормалев Д.А. Система извлечения информации из текстов INEX / Д.А. Кормалев, Е.П. Куршев, Е.А. Сулейманова, И.В. Трофимов // Девятая Национальная конференция по искусственному интеллекту с международным участием КИИ–2004: Труды конференции. – М.: Физматлит, 2004. – Т.3. – С. 908–915.
[25] Осовский С. Нейронные сети для обработки информации. – М.: Финансы и статистика, 2002. – 344 с.: ил. С. 46-89.
[26] Розенблатт, Ф. Принципы нейродинамики: Перцептроны и теория механизмов. — М.: Мир, 1965. — 480 с.
[27] Симаков К.В. Модель извлечения знаний из естественно-языковых текстов / А.М. Андреев, Д.В. Березкин, К.В. Симаков // Информационные технологии. – 2007. – №12. – С. 57–63.
[28] Симаков К.В. Метод обучения модели извлечения знаний из естественно-языковых текстов / А.М. Андреев, Д.В. Березкин, К.В. Симаков // Вестник МГТУ. Приборостроение.–2007. – №3.– С. 75–94.

Machine learning in information extraction having etalon database

Alexeev S, Morozov V, Simakov K

We describe a special case of task of information extraction from texts when a whole database of objects to extract is already exists. Such database includes only canonical representations of objects, so the task is to recognize them by their non-canonical descriptions in texts. To disambiguate the result of such recognition we research, test and compare a range of machine learning methods. The result of such comparison is also described.