Извлечение информации из текста с автоматическим построением правил

© П.А. Прокофьев, В. Г. Васильев

ООО «ЛАН-ПРОЕКТ»
p_prok@mail.ru

Аннотация

Методы извлечения информации из текстов, как правило, не дают экспертам четкого понимания того, какие факторы влияют на принятие решений при обработке информации. Желательно, чтобы правила извлечения были описаны на понятном экспертам языке и могли быть изменены экспертами вручную. В данной работе предлагается подход, основанный на построении дискретных процедур распознавания. Настройка процедур распознавания сопровождается автоматическим построением правил извлечения фрагментов.

Введение

Задача извлечения информации из текста относится к разряду тех, для которых практически невозможно построить математическую модель в общепринятом смысле. Частным случаем задачи извлечения информации из текста является отнесение фрагментов в тексте к одному или нескольким заранее определенным классам.

В работе [9] извлечение информации использовалось для разрешения неоднозначностей выделения географических объектов в текстах. Слова и словосочетания в текстах распределялись по классам: страна, область, район, город и прочие. Используемые в работе [9] методы строили модели, анализ которых для экспертов достаточно сложен.

Другой подход, основанный на написании правил экспертами вручную, показывает, что правила быстро разрастаются и становятся плохо понятными самим экспертам.

В работах [1,4] используются методы автоматического построения логических правил на этапе настройки классификатора. Правила позволяют отнести целый текст или его части к определенной тематике. Язык описания правил достаточно узок. Все это делает проблематичным использование этих методов в задаче извлечения информации в текстах.

В работе [2] предлагается метод формального описание правил извлечения информации. По правилам строятся наборы фрагментов текстов. Подобный подход разрабатывается в данной работе, но в большей степени уделяется внимание автоматизации процесса построения правил.

В рамках данной работы предлагается формальное описание языка правил извлечения фрагментов текста и метод автоматического построения правил при настройке алгоритмов классификации, используемых при извлечении информации в тексте. Алгоритмы классификации строятся с помощью дискреционных процедур распознавания по прецедентам, описанных в работах [6,7,8].

Методы рассмотренные в настоящей работе оценивались при решении задачи разрешения неоднозначностей при географической привязке текстов.

Модель текстов

Определение 1. *Текстами* будем называть конечные последовательности слов: $\tau = (\tau_1, \ldots, \tau_L)$, где τ_i – слова, $\tau_i \in W$, $i \in \{1, \ldots, L\}$, W – множество допустимых слов, $L = L(\tau)$ – длина текста. Текст нулевой длины обозначим Λ. Обозначим множество всех текстов T.

Замечание 1. В настоящей работе словом текста является совокупность исходной строки в тексте, границ слова в тексте, набора морфологических форм слова и набора дескрипторов.

Определение 2. *Фрагментом текста* τ называется совокупность трех объектов: τ, i, j, для которых выполняется условие $1 \le i \le j \le L(\tau)$. Фрагмент текста будем обозначать $\tau[i, j]$, где i – *начало фрагмента*, j – *конец фрагмента*. Обозначим $\mathrm{F}(\tau)$ – множество всех фрагментов текста τ, F – множество всех фрагментов текстов. В случаях, когда нет необходимости указывать текст и границы фрагмента, для обозначения фрагментов текстов будем использовать заглавные латинские буквы.

Для фрагмента $S = \tau[i, j]$ число $j - i + 1$ будем называть длиной фрагмента и обозначать $|S|$.

Труды 13[й] Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» - RCDL'2011, Воронеж, Россия, 2011.

Если фрагмент $S_1 = \tau[i_1, j_1]$ включает в себя фрагмент $S_2 = \tau[i_2, j_2]$, то есть $i_1 \le i_2 \le j_2 \le j_1$, то будем писать S_1 с S_2.

Если фрагмент $S_1 = \tau[i_1, j_1]$ расположен слева от фрагмента $S_2 = \tau[i_2, j_2]$, то есть $i_1 \le j_1 < i_2 \le j_2$, то будем писать $S_1 < S_2$.

Для фрагментов $S_1 = \tau[i_1, j_1]$, $S_2 = \tau[i_2, j_2]$, таких что $S_1 < S_2$ и $n \le i_2 - j_1 \le m$, будем писать $S_1 <_{n,m} S_2$.

Правила извлечения фрагментов

Определение 3. *Правилом извлечения фрагментов* (или просто правилом) будем называть отображение q, ставящее в соответствие любому тексту τ конечный набор фрагментов этого текста:

$$q(\tau) = \{\tau[i_1, j_1], \ldots, \tau[i_c, j_c]\} \subset \mathrm{F}(\tau).$$

Рассмотрим несколько правил, которые будут использоваться в настоящей работе:

1) $q^*(\tau) = \mathrm{F}(\tau)$ возвращает все фрагменты текста;

2) $q^{[n,m]}(\tau) = \{f \in \mathrm{F}(\tau) \mid n \le |f| \le m\}$, возвращает фрагменты ограниченной длины; $q^{[n]} = q^{[n,n]}$;

3) $q^{(g,\alpha)}(\tau) = \{\tau[i, j] \mid g(\alpha, (\tau_i, \ldots, \tau_j)) = 1\}$,

возвращает фрагменты, последовательность слов в которых делает истинным предикат $g : A \times \mathrm{T} \to \{0,1\}$, зависящий от параметра, значение которого фиксируется значением $\alpha \in A$. Использование такой конструкции позволяет строить правила, которые проверяют наличие дескрипторов у слов, сравнивают слова с учетом морфологии, или проверяют удовлетворение фрагментов текста регулярным выражениям.

Чтобы задание правил сделать конструктивным, рассмотрим ряд операций:

1) пересечение:

$$(q_1 \Diamond q_2)(\tau) = q_1(\tau) \cap q_2(\tau);$$

2) последовательность с ограничением на расстояние:

$$(q_1 \mathrm{W}_{n,m} q_2)(\tau) = \{S_1 \Delta S_2 \mid S_i \in q_i(\tau), S_1 <_{n,m} S_2\};$$

3) унарная и бинарная операции содержания фрагментов:

$$(\lhd q)(\tau) = \{S \mid \exists S_1 \in q(\tau), S_1 \text{ с } S\},$$

$$q_1 \lhd q_2 = q_1 \Diamond (\lhd q_2),$$

возвращает фрагменты q_1, содержащиеся в некотором фрагменте q_2;

4) унарная и бинарная операции префиксного условия:

$$(\mathrm{W}^{\leftarrow} q)(\tau) = \{S \mid \exists S_1 \in q(\tau), S_1 < S\},$$

$$q_1 \mathrm{W}^{\leftarrow} q_2 = (\mathrm{W}^{\leftarrow} q_1) \Diamond q_2$$

возвращает фрагменты результата q_2, справа от фрагментов результата q_1;

5) аналогично задается суффиксное условие:

$$(\mathrm{W}^{\rightarrow} q)(\tau) = \{S \mid \exists S_1 \in q(\tau), S < S_1\},$$

$$q_1 \mathrm{W}^{\rightarrow} q_2 = q_1 \Diamond (\mathrm{W}^{\rightarrow} q_2);$$

6) аналогично 5 и 6 задаются условия с ограничением на расстояние: $q_1 \mathrm{W}^{\rightarrow}_{n,m} q_2$ и $q_1 \mathrm{W}^{\leftarrow}_{n,m} q_2$.

Задача классификации фрагментов текста

В настоящей работе будет рассмотрен частный случай задачи извлечения информации в тексте, когда необходимо классифицировать фрагменты в тексте по нескольким заранее определенным классам.

Пусть известно, что множество фрагментов текстов F представимо в виде объединения непересекающихся классов $\mathrm{K}_1, \ldots, \mathrm{K}_m$. Имеется конечный набор фрагментов $X = \{S_1, S_2, \ldots, S_l\} \subset \mathrm{F}$, для которых известна их принадлежность к классам. Требуется для произвольного фрагмента определить класс, к которому он принадлежит.

Введем обозначения $K_i = \mathrm{K}_i \cap X$ – конечный набор объектов выборки из класса K_i, $\overline{K_i} = X \setminus K_i$ – объекты выборки вне класса K_i.

Признаки фрагментов текстов

Определение 4. *Признаком* фрагмента называется любое отображение, ставящее в соответствие фрагменту S определенное значение $\alpha = f(S)$.

Признак фрагмента может быть задан с помощью правила q следующим образом:

$$f^{(q)}(\tau[i, j]) = \begin{cases} 1, \text{если } \tau[i, j] \in q(\tau), \\ 0, \text{иначе.} \end{cases}$$

Если задан набор различных правил $R = \{q_1, \ldots, q_l\}$ и зафиксирован некоторый их порядок, то любому фрагменту S соответствует двоичный вектор значений признаков для правил: $f^{(R)}(S) = \left(f^{(q_1)}(S), \ldots, f^{(q_l)}(S)\right)$.

Вектор значений признаков, соответствующих поднабору правил $H = \{q_{i_1}, \ldots, q_{i_r}\} \subset R$, для фрагмента S будем называть подописанием и обозначать $(S, H) = \left(f^{(q_{i_1})}(S), \ldots, f^{(q_{i_r})}(S)\right)$.

Близость подописаний фрагментов S и S' по поднабору правил H будем оценивать величиной:

$$B(S, S', H) = \begin{cases} 1, f^{(q_{i_u})}(S) = f^{(q_{i_u})}(S'), \forall u \in 1, \ldots, r, \\ 0, \text{иначе.} \end{cases}$$

Построение набора информативных признаков

Пусть имеется некоторый поднабор правил H. Для подописания фрагмента $(S, H), S \in K_i$ введем следующую величину:

$$\mu_i(S,H) = \frac{1}{|K_i|}\sum_{S'\in K_i} B(S,S',H) - \frac{1}{|\overline{K}_i|}\sum_{S'\in \overline{K}_i} B(S,S',H).$$

В работе [6] предлагается использовать величину $\mu_i(S,\{q\})$ для характеристики информативности значения признака $f^{(q)}$, принимаемого на фрагменте S из класса K_i.

Несколько обобщим это понятие для характеристики информативности поднабора признаков. Интуитивно понятно, что набор признаков информативен для класса, если информативно хотя бы одно подописание для поднабора в этом классе. Тогда величина

$$\mu_i(H) = \max_{S\in K_i}\mu_i(S,H),$$

характеризует информативность поднабора H для класса K_i.

Информативность поднабора $H \subset R$ также характеризуется максимальной информативностью по поднаборам ограниченной длины, содержащих H:

$$\mu_i^{(r)}(H,R) = \max_{H'\subset R, H\subset H', |H'|\le r} \mu_i(H').$$

Информативность всего набора в среднем для класса K_i может быть оценена величиной:

$$\overline{\mu}_i^{(r)}(R) = \frac{1}{l}\sum_{j=1}^{l}\mu_i^{(r)}(\{q_j\},R).$$

Информативность всего набора для выборки может быть оценена величиной:

$$\overline{\mu}^{(r)}(R) = \min_{i\in\{1,\dots,m\}} \overline{\mu}_i^{(r)}(R),$$

так как для каждого класса в наборе должны содержаться информативные признаки.

Введем еще одну характеристику для подописания фрагмента $(S,H), S\in K_i$:

$$\nu_i(S,H) = \sum_{S'\in \overline{K}_i} 1 - B(S,S',H),$$

равную числу фрагментов в других классах выборки, имеющих отличное от (S,H) подописание.

Значение

$$\nu(R) = \frac{1}{\Theta}\sum_{i=1}^{m}\sum_{S\in K_i}\nu_i(S,R), \text{ где } \Theta = \sum_{i=1}^{m}|K_i|\left|\overline{K}_i\right|$$

характеризует то, как набор R разделяет классы выборки.

Критерием для выбора набора правил в настоящей работе является максимизация значения функционала

$$\lambda(R) = \nu(R)\cdot\overline{\mu}^{(2)}(R).$$

Критерий отдает предпочтение коротким наборам, хорошо разделяющим классы выборки и состоящим из правил информативных по отдельности и по парам.

Изначально набор правил строится из простых (базовых) правил определенного вида. Базовые правила строятся исходя из контекста фрагментов обучающей выборки. В настоящей работе использовался алгоритм 1 для формирует набор базовых правил по контексту фрагмента. Алгоритм 1 строит набор, состоящий из правил с ограничением трех типов: на сам фрагмент, на префикс и на суффикс. Ограничения задаются предикатами. В настоящей работе использовались предикаты следующих типов:

1) $g_1(\omega,(\tau_i)) = 1$, только если в тексте τ i-е слово равно ω;

2) $g_2(\omega,(\tau_i)) = 1$, только если в тексте τ исходная морфологическая форма i-го слова равна ω;

3) $g_3(d,(\tau_i)) = 1$, только если в тексте τ у i-го слова есть дескриптор d, например, `$FirstUp` (слово с большой буквы), `$SentEnd` (конец предложения), `$Verb` (глагол) и другие; всего в настоящей работе использовалось 43 дескриптора;

4) $g_4(D,(\tau_i)) = 1$, только если в тексте τ исходная морфологическая форма i-го слова принадлежит множеству D (словарю); в настоящей работе использовались словари географических названий, имен, фамилий, отчеств, аббревиатур, различных типов предлогов, родовые словари географических объектов.

Из всех построенных для контекстов фрагментов правил выбирается оптимальный в некотором смысле поднабор. В настоящей работе использовался алгоритм 2, в основе которого лежит стратегия **Add-Del**, заключающаяся в последовательном добавлении в набор (см. шаг 6), а затем удалении из набора нескольких правил (см. шаг 13) так, чтобы найти набор, на котором функционал качества $\lambda(R)$ имел бы оптимальное значение.

Дискретные процедуры классификации

Зафиксируем порядок отобранных правил $R = \{q_1,\dots,q_l\}$ и рассмотрим множества векторов значений признаков $\tilde{K}_i = f^{(R)}(K_i)\setminus f^{(R)}(\overline{K}_i)$. Множество $\tilde{X} = \tilde{K}_1$ т … т $\tilde{K}_m$ будем далее называть обучающей выборкой.

В работах [6,7] описаны методы построения дискретных процедур распознавания (классификации). Приведем здесь основные понятия необходимые для изложения результатов настоящей работы. Опишем дискреционные процедуры классификации на языке логических функций.

Определение 5. Будем называть *элементарным классификатором* элементарную конъюнкцию $x_{i_1}^{\alpha_1}\cdot\ldots\cdot x_{i_r}^{\alpha_r}$ от переменных $x_{i_1},\dots,x_{i_r}$, где

$$x^{\alpha} \equiv \begin{cases} 1, x = \alpha, \\ 0, \text{ иначе.} \end{cases}$$

Определение 6. Если задан элементарный классификатор $c = x_{i_1}^{\alpha_1}\cdot\ldots\cdot x_{i_r}^{\alpha_r}$, то для вектора $\vec{\beta} = (\beta_1,\dots,\beta_l)$ будем обозначать

$$B(c,\vec{\beta}) = \beta_{i_1}^{\alpha_1}\cdot\ldots\cdot\beta_{i_r}^{\alpha_r}.$$

Алгоритм 1. Построения набора «базовых» признаков из контекста фрагмента

Вход: фрагмент $\tau[u,v]$; простое правило q_0, которому удовлетворяет $\tau[u,v]$; множество предикатов $G=\{g_1,\ldots,g_d\}$, $g_i: A_i\times\mathbb{T}\to\{0,1\}$, $\forall i\in\{1,\ldots,d\}$ для построения базовых правил; максимальные расстояния $n^{(p)}$ и $n^{(s)}$ до префиксного и суффиксного условий соответственно.

Выход: B — набор базовых правил, извлеченных из контекста фрагмента.

1: инициализация: $B:=\varnothing$;
2: **для всех** $g_h\in G$
правила по предикату на фрагменте:
3: **для всех** $\tau[i,j]\sqsubset\tau[u,v]$
4: **для всех** $\alpha: g_h(\alpha,(\tau_i,\ldots,\tau_j))=1$
5: $B:=B\cup\left\{q^{(g_h,\alpha)};q^{(g_h,\alpha)}\lhd q_0\right\}$
правила по предикату перед фрагментом:
6: **для всех** $\tau[i,j]\sqsubset\tau[u-n^{(p)},u-1]$
7: **для всех** $\alpha: g_h(\alpha,(\tau_i,\ldots,\tau_j))=1$
8: $B:=B\cup\left\{q^{(g_h,\alpha)}\square^{\leftarrow}_{n,n}q_0;q^{(g_h,\alpha)}\square^{\leftarrow}q_0\right\}$, где $n=u-v^{(p)}$;
правила по предикату после фрагмента:
9: **для всех** $\tau[i,j]\sqsubset\tau[v+1,v+n^{(s)}]$
10: **для всех** $\alpha: g_h(\alpha,(\tau_i,\ldots,\tau_j)=1$
11: $B:=B\cup\left\{q_0\square^{\rightarrow}_{n,n}q^{(g_h,\alpha)};q_0\square^{\rightarrow}q^{(g_h,\alpha)}\right\}$, где $n=u^{(s)}-v$;

Будем говорить, что вектор $\vec{\beta}$ удовлетворяет элементарному классификатору c, если $B(c,\vec{\beta})=1$.

Дискретная процедура классификации A состоит из множеств элементарных классификаторов $C^A(\tilde{K}_1),\ldots,C^A(\tilde{K}_m)$ для каждого класса, а также весов элементарных классификаторов $\gamma(c)$, для всех $c\in C^A(\tilde{K}_i)$.

В работе [6] описаны процедуры состоящие из элементарных классификаторов, обладающих определенными свойствами.

Определение 7. Элементарный классификатор называется *представительным набором* для класса $\tilde{K}_i$, если ему не удовлетворяет ни один вектор из $\tilde{X}\setminus\tilde{K}_i$. Процедуры, состоящие только из представительных наборов, называются *голосованием по представительным наборам*.

Определение 8. Элементарный классификатор называется *антипредставительным набором* для класса $\tilde{K}_i$, если ему не удовлетворяет ни один вектор из $\tilde{K}_i$. Процедуры, состоящие только из антипредставительных наборов, называются *голосованием по антипредставительным наборам*.

В настоящей работе также рассматриваются процедуры, состоящие из классификаторов обоих типов. Именно такие процедуры имеют наилучшие показатели.

Алгоритм 2. Построение набора «базовых» признаков для классификации фрагментов

Вход: Обучающая выборка $X=K_1\sqcup\ldots\sqcup K_m$; параметры $\Delta_1\geqslant 0$, $\Delta_2\geqslant 0$; максимизируемый функционал качества $\lambda(R)$.

Выход: Набор признаков R_{max} с наибольшим среди построенных значением $\lambda(R_{max})$.

1: инициализация: $R_{max}:=\varnothing$; $\lambda_{max}:=\lambda(R_{max})$; $R:=\varnothing$ — текущий набор правил; $U:=\varnothing$ — множество проверенных правил;
2: **для всех** K_i в порядке увеличения $|K_i|$
начать перебирать фрагменты класса:
3: **для всех** $\tau[u,v]\in K_i$
4: построить по алгоритму 1 набор базовых правил B для $\tau[u,v]$;
5: запомнить текущее качество $\lambda_0:=\lambda(R)$;
добавить новые правила, контролируя качество:
6: **для всех** $q\in B\setminus U$ в порядке уменьшения $\mu(\{q\})$
7: **если** $\lambda(R)-\lambda(R\cup\{q\})<\Delta_1$ **то**
8: добавить $R:=R\cup\{q\}$; $U:=U\cup\{q\}$;
9: **если** $\lambda_{max}<\lambda(R)$ **то**
10: $R_{max}:=R$; $\lambda_{max}:=\lambda(R)$;
11: **если** $\lambda_0-\lambda(R)>\Delta_2$ **то**
прервать цикл;
12: запомнить текущее качество $\lambda_0:=\lambda(R)$;
удалить «лишние» правила после обработки фрагментов класса:
13: **для всех** $q\in R$ с уменьшением $\mu(\{q\})$
14: **если** $\lambda(R)-\lambda(R\setminus\{q\})<\Delta_1$ **то**
15: удалить $R:=R\setminus\{q\}$;
16: **если** $\lambda_{max}<\lambda(R)$ **то**
17: $R_{max}:=R$; $\lambda_{max}:=\lambda(R)$;
18: **если** $\lambda_0-\lambda(R)>\Delta_2$ **то**
прервать цикл;

При классификации вектора $\vec{\beta}$ для каждого класса $\tilde{K}_i$ вычисляются значения функции голосования. Например, при голосовании по представительным наборам:

$$\Gamma_1(\vec{\beta},\tilde{K}_i)=\sum_{c\in C^A(\tilde{K}_i)}\gamma(c)B(c,\vec{\beta}).$$

Вектор относится к классу, для которого значение функции голосования максимально.

Поиск элементарных классификаторов

В работе [6] описан способ нахождения представительных и антипредставительных наборов, как элементарных конъюнкций допустимых для частично определенной логической функции. Такие элементарные конъюнкции называются *импликантами* функции. Рассмотрим этот способ на примере голосования по представительным наборам.

Пусть для класса $\tilde{K}_i$ частично определена на $\tilde{X}$ функция

$$u^{(\tilde{K}_i,\tilde{X})}(\vec{\beta}) = \begin{cases} 1, \vec{\beta} \in \tilde{K}_i \\ 0, \vec{\beta} \in \tilde{X} \setminus \tilde{K}_i. \end{cases} \quad (1)$$

Доопределяя функцию $u^{(\tilde{K}_i,\tilde{X})}$ на множество $\{0,1\}^l$, получаем функцию

$$U^{(\tilde{K}_i,\tilde{X})}(\vec{\beta}) = \begin{cases} 0, \vec{\beta} \in \tilde{X} \setminus \tilde{K}_i \\ 1, \text{иначе}. \end{cases} \quad (2)$$

Теоретически импликанты можно получить при нахождении сокращенной ДНФ доопределенной функции. Сокращенная ДНФ находится методами, приведенными в книге [10].

Если $\tilde{X} \setminus \tilde{K}_i = \left\{\vec{\beta}^{(1)}, \ldots, \vec{\beta}^{(h)}\right\}$, то

$$U^{(\tilde{K}_i,\tilde{X})} = \overline{\delta_1} \wedge \ldots \wedge \overline{\delta_h},$$

где конъюнкции $\delta_j = x_1^{\beta_1^{(j)}} \cdot \ldots \cdot x_l^{\beta_l^{(j)}}$ соответствуют векторам $\vec{\beta}^{(j)}$, $j = 1, \ldots, h$. Тогда для нахождения сокращенной ДНФ функции (2) необходимо перемножить все элементарные дизъюнкции, получающиеся после отрицания, а затем применить правило поглощения: $\psi\phi \equiv \phi$, где ϕ и ψ – элементарные конъюнкции.

Поскольку нас интересуют только импликанты частично определенной функции (1), то мы можем оставить только те, которые истинны хотя бы для одного $\vec{\beta} \in \tilde{K}_i$. Фильтровать таким образом импликанты мы можем после каждого перемножения скобок. Однако в настоящей работе набор признаков превышает 50, число перемножаемых дизъюнкций для некоторых классов превышает 10^3, а при перемножении уже 5 скобок получается более 10^7 конъюнкций, в том числе при фильтрации, время нахождения импликантов по такому алгоритму становится неприемлемым даже на мощных компьютерах.

В работе [6] предлагается искать только элементарные классификаторы, обладающие определенными свойствами.

Определение 9. Элементарный классификатор c будем называть p-представительным набором для класса $\tilde{K}_i$, если ему удовлетворяет не менее p векторов из $\tilde{K}_i$. Элементарный классификатор c будем называть r-антипредставительным набором для класса $\tilde{K}_i$, если ему удовлетворяет не более r векторов из $\tilde{X} \setminus \tilde{K}_i$.

В работе [6] говорится, что варьируя параметры p и r, можно существенно снизить влияние шумящих признаков, а также сократить сложность нахождения элементарных классификаторов однако нет рекомендаций по подбору этих параметров. В настоящей работе предлагается адаптивный алгоритм нахождения элементарных классификаторов (алгоритм 3), удовлетворяющих порогам p и r, изменяющимся в процессе вычислений по критерию допустимой сложности вычислений. На шаге 4 проверяется сколько умножений необходимо совершить. Если число умножений превосходит заданный порог, то применяем различные правила к скобкам, уменьшая число конъюнкций в них.

Замечание 3. Значительно сократить время работы алгоритма 3 удалось при использовании алгоритмов включающего (дескрипторного) поиска, подробно проанализированных в книге [5].

Выбор весов элементарных классификаторов

В работе [6] предлагается разбивать обучающую выборку на базовую и контрольную. Наборы элементарных классификаторов строятся по базовой выборке, а веса вычисляются исходя из классификации контрольной выборки.

Пусть $\delta'(c)$ – число объектов контрольной выборки, за которые элементарный классификатор класса $\tilde{K}_i$ проголосовал правильно, $\delta''(c)$ – число неправильных голосований. В настоящей работе используется следующий способ вычисления веса.

$$\gamma(c) = \begin{cases} \frac{1}{N} \frac{\delta'(c)}{\delta'(c) + \delta''(c)}, \delta(c) + \delta''(c) > 0, \\ \frac{1}{N} \frac{1}{2}, \text{иначе}, \end{cases} \quad (3)$$

где N – нормирующий множитель, такой что $\sum_{c \in C^A(\tilde{K}_i)} \gamma(c) = 1$.

Другой подход вычисления весов может быть основан на других обучаемых методах классификации, например, **MaxEnt** [3]. В настоящей работе построенные по базовой выборке элементарные классификаторы для всех классов объединяются в один набор и рассматриваются как признаки. Далее обучение и классификация осуществляется методом MaxEnt.

Эксперименты

Метод, предложенный в работе [9] географической привязки текстов, заключался в классификации фрагментов, которые были найдены в географическом справочнике. Слова классифицировались по типам географических объектов, чтобы снять неоднозначность, связанную с неуникальностью имен (объекты разных типов имеют одинаковые названия).

Предложенный в настоящей работе подход был протестирован на этой задаче. Размеченный корпус состоит из 846 текстов, содержащих 372367 слов (в то числе знаков пунктуации). В текстах 5431 фрагментов было отнесено к одному из 8 типов географических объектов. При поиске в географическом справочнике было найдено 6408 «подозрительных» фрагментов, то есть 977 фрагментов были отнесены к классу «другие».

Выбор базовых правил осуществлялся по всей обучающей выборке. Было отобрано 56 правил.

Алгоритм 3. Поиск импликантов частично определенной логической функции с адаптацией по сложности вычислений

Вход: A — множество истинности функции; B — множество ложности функции; M — порог максимального числа умножений элементарных конъюнкций при перемножении двух скобок. p — начальное условие p-представительности импликантов; r — начальное условие r-антипредставитости импликантов;

Выход: Набор импликантов со свойствами p-представительности и r-антипредставитости.

1: инициализация: $D = \{(x_1^{\bar{\beta}_{i_1}} \vee \ldots \vee x_l^{\bar{\beta}_{i_l}}), \boldsymbol{\beta} \in B\}$ — множество перемножаемых скобок;
дополнительные обозначения:
число конъюнкций в скобке $d \in D$ обозначается $|d|$;
для D две скобки с наименьшим числом конъюнкций обозначаются $min_1(D), min_2(D)$;
сложность очередного перемножения обозначается $c(D) = |min_1(D)||min_2(D)|$;
2: удалить из скобок в D не 1-представительные для A конъюнкции;
3: **пока** $|D| > 1$
проверить сложность перемножения двух наименьших скобок;
4: **пока** $c(D) > M$
5: применить к скобкам правила поглощения, пока это необходимо;
6: удалить не p-представительные конъюнкции в скобках, пока это необходимо;
7: удалить не r-антипредставительные конъюнкции в скобках, пока это необходимо;
8: **если** $c(D) > M$ **то**
9: увеличить p; уменьшить r;
10: после нескольких попыток **прервать цикл** уменьшения сложности;
11: $d := min_1(D) \cdot min_2(D)$; удалить из d не 1-представительными для A конъюнкции;
12: $D := D \setminus \{min_1(D), min_2(D)\} \cup \{d\}$;
13: применить к оставшейся скобке правила поглощения и удалить не p-представительные и не r-антипредставительные для A;

Среди автоматически отобранных правил встречались вполне ожидаемые, например, приведенные в таблице 1. Правила в таблице написаны на специальном языке так, чтобы эксперты, знающие синтаксис языка, могли понять и исправить при необходимости правила. Строка «`@ОБЛАСТЬ`» соответствует правилу $q^{(g_4, D_{обл.})}$, где $D_{обл.}$ – словарь названий областей России. Строка «`#1 :1? РАЙОН`» соответствует правилу $q^{[1]} \mathrm{W}_{1,1}^{\rightarrow} q^{(g_2, РАЙОН)}$.

Таблица 1. Примеры отобранных автоматически «базовых» правил

Правило (q)	$\mu(\{q\})$
`@ОБЛАСТЬ`	0,98
`#1 :1? РАЙОН`	0,36
`#1 :1? КРАЙ`	0,82
`ЕДИНАЯ РОССИЯ`	0,05

Таблица 2. Оценка качества методов

Метод	P	R	F	e
ME	42,5%	37,6%	35,2%	17,4%
Γ_1	76,2%	65,8%	63,3%	15,6%
Γ_2	38,3%	24,8%	21,7%	27,2%
$\Gamma_1 + ME$	76,1%	59,8%	54,9%	22,2%
$\Gamma_2 + ME$	30,0%	21,3%	19,9%	25,5%
$\Gamma_1 + \Gamma_2$	**77**,1%	**72**,6%	**69**,6%	**11**,2%

Таблица 3. Оценка качества для хорошего класса «`Область`»

Метод	P	R	F	e
ME	61,3%	86,9%	71,9%	7,7%
Γ_1	94,3%	98,0%	96,2%	0,9%
Γ_2	0,0%	0,0%	0,0%	12,0%
$\Gamma_1 + ME$	95,4%	98,8%	97,1%	0,7%
$\Gamma_1 + \Gamma_2$	94,3%	98,0%	96,2%	0,3%

В настоящей работе были оценены показатели качества для следующих методов:

1) метод классификации MaxEnt (ME);

2) по представительным наборам (Γ_1);

3) по антипредставительным наборам (Γ_2);

4) MaxEnt с использование представительных наборов как признаков ($\Gamma_1 + ME$);

5) MaxEnt с использование антипредставительных наборов как признаков ($\Gamma_2 + ME$);

6) голосование по представительным и антипредставительным наборам ($\Gamma_1 + \Gamma_2$);

Для каждого метода проводилось по 3 итерации. При обучении на каждой итерации выборка случайно делилась на тестовую (30%) и обучающую (70%). При оценке методов Γ_1 и Γ_2 обучающая выборка разбивалась на базовую (50%) и проверочную (50%) для вычисления весов по формуле (3).

При оценке каждого метода вычислялись показатели точности (P), полноты (R), F-меры (F), а также процент ошибки (e) для каждого класса, а также макро-усреднения этих параметров. Макро-усреднения оценок приведены в таблице 2.

Приведем также показатели для «хорошего» («`Область`» в таблице 3) и «плохого» («`Республика`» в таблице 4) классов.

Таблица 4. Оценка качества для плохого класса «Республика»

Метод	P	R	F	e
ME	60,7%	89,0%	72,2%	35,2%
Γ_1	51,7%	100,0%	68,2%	42,1%
Γ_2	97,8%	12,9%	30,0%	54,2%
$\Gamma_1 + MaxEnt$	98,0%	23,7%	38,2%	50,5%
$\Gamma_1 + \Gamma_2$	73,7%	94,6%	82,9%	12,8%

Таблица 5. Примеры «сложных» правил по представительным наборам

Текст правила	Классификатор
`@ОБЛАСТЬ :1! РАЙОН`	$x_1^1 x_2^0$
`@ОБЛАСТЬ :1! {$Verb}`	$x_1^1 x_3^0$
Переменная	Правило
x_1	$q^{(g_4, D_{\text{обл.}})}$
x_2	$q^{[1]} \square_{1,1}^{\rightarrow} q^{(g_2, \text{РАЙОН})}$
x_3	$q^{[1]} \square_{1,1}^{\rightarrow} q^{(g_3, \text{Verb})}$

Худшие результаты в среднем показал метод голосования по антипредставительным наборам. Для больших классов были построены длинные антипредставительные наборы, которые на тестовой выборке голосовали очень редко, и были получены показатели $P = 0$ и $R = 0$.

Лучшие результаты показал метод голосования по представительным и антипредставительным наборам, в котором складываются голоса классификатором из процедур Γ_1 и Γ_2.

Кроме показателей качества приведем примеры более сложных правил, которые могут быть построены по представительным наборам, найденным при обучении. В таблице 5 приведены несколько правил по элементарным классификаторам длины 2 для класса «`Область`». В строке «`@ОБЛАСТЬ :1! РАЙОН`» оператор «`:1!`» означает отрицание суффиксного правила.

Анализ элементарных классификаторов может быть автоматизирован для построения более сложных правил. Этому вопросу будут посвящены дальнейшие исследования.

Выводы

В работе дается формальное описание конструкции правил извлечения фрагментов текста. Показывается принципиальная возможность представления элементарных классификаторов в виде правил. Дальнейшие исследования будут направлены на следующие моменты:

1) расширение и строгое формальное описание языка правил;

2) разработка методов формирования коротких элементарных классификаторов;

3) разработка методов построения сложных правил по элементарным классификаторам;

4) исследование сложности алгоритмов автоматического построения правил.

Литература

[1] Junker M., Abecker A. Learning Complex Patterns for Document Categorization // AAAI-98/ICML Workshop on Learning for Text Categorization. Madison, Wisconsin, USA, 1998.

[2] Reiss F.,Raghavan S., Krishnamurthy R., Zhu H., Vaithyanathan S. An Algebraic Approach to Rule-Based Information Extraction // ICDE. Cancun, Mexico, 2008.

[3] Ratnaparkhi A. Maximum Entropy Models for Natural Language Ambiguity resolution. PHD thesis, Univ. of Pennsylvania, 1998.

[4] Агеев М. С. Методы автоматической рубрикации текстов, основанные на машинном обучении и знаниях экспертов. – Диссертация на соискание ученой степени к.ф.-м.н. – М.: МГУ, 2004.

[5] Гасанов Э. Э., Кудрявцев В. Б. Теория хранения и поиска информации. – М.: ФИЗМАТЛИТ, 2002. – 288 с.

[6] Дюкова Е. В., Песков Н. В. Построение распознающих процедур на базе элементарных классификаторов // Математические вопросы кибернетики / Под ред. О.Б. Лупанов. – М.: Физматлит, 2005. – Т. 14.

[7] Дюкова Е. В. Дискретные (логические) процедуры распознавания: принципы конструирования, сложность реализации и основные модели. Учебное пособие для студентов математических факультетов педвузов. – М.: Изд-во «Прометей», 2003. – 29 с.

[8] Песков Н. В. Поиск информативных фрагментов описаний объектов в задачах распознавания. – Диссертация на соискание ученой степени к.ф.-м.н., М.: ВЦ РАН. – 2004.

[9] Прокофьев П. А. Использование методов извлечения информации при географической привязке текстов на русском языке // Электронные библиотеки: Перспективные Методы и Технологии, Электронные коллекции (RCDL). Петрозаводск, 2009.

[10] Дискретная математика и математические вопросы кибернетики / Под ред. С.Б. Яблонского, О.Б. Лупанова, М.: «Наука», 1974. 312 с.

Information Extraction from Texts with Automatic Construction of Rules

© P.A. Prokofyev, V.G. Vasilyev

Methods for information extraction from texts, generally, do not give experts a clear understanding of what factors affect the decision making in information processing. It is desirable that the extraction rules could be described in plain language and the experts should be able to change them manually. In this paper we propose an approach based on constructing the discrete recognition procedures. Setting up the automatic recognition procedures is followed by construction of the rules for extraction of the fragments.