Модель извлечения графематических дескрипторов в системе обработки вьетнамского языка[1]

Ле Чунг Хъеу
Санкт-Петербургский государственный университет
vkhhieukien@yahoo.com

В работе рассматривается модель извлечения графематических дескрипторов в системе обработки вьетнамского языка. Модель использует метод, основанный на сопоставлении образцов в большом текстовом массиве данных, для реализации следующих основных функций: (i) извлечение графематических дескрипторов из текстов (знаков пунктуации, цифровых комплексов, формул, собственных имен, сокращений, и т. п.); (ii) определение границ предложений, фраз. Ключевым элементом модели является набор правил извлечения. Для эксперимента данные были взяты из 283 992 онлайн статей, в которых около 15 000 000 фраз.

Ключевые слова: извлечение графематических дескрипторов, распознавание фактов на вьетнамском языке, графематический анализ, метод, основанный на сопоставлении образцов.

1. Введение

Автоматизированная обработка печатных документов — одна из основных задач формирующихся систем искусственного интеллекта [1]. Типичная *задача извлечения фактов* заключается в обработке текста на естественном языке с целью извлечения заданных элементов. На входе процесса извлечения — слабоструктурированный или неструктурированный текст на естественном языке; на выходе — заполненные структуры данных, позволяющие проводить дальнейшую автоматическую или ручную обработку информации.

В настоящий момент не существует обучаемых моделей извлечения фактов из текстов, не привязанных к конкретному языку, обладающих должным качеством извлечения [2]. Необходимо иметь возможность как точной ручной настройки модели, так и автоматической — на основе обучающих примеров. Этому требованию в большей степени удовлетворяют символические модели.

Задача извлечения графематических дескрипторов — выделение различных нестандартных элементов текста и присваивание им соответствующих графематических дескрипторов (например, знаков пунктуации, цифровых комплексов, собственных имен, сокра-

щений и т. п.). Элементами графематического дескриптора являются:

- структурные элементы текста — заголовки, абзацы, примечания, предложения из входного текста;
- различные элементы текста, не являющиеся слогами (цифры и числа, даты в цифровых форматах, буквенно-цифровые комплексы, цифрово-знаковые комплексы и т. п.);
- собственные имена, названия, аббревиатуры;
- иностранные лексемы.

Основные трудности задачи:

1. Распознавание случаев использования знаков препинания в иных целях, а не в качестве разделителя фразы. Например, точка в сокращенном слове (“*TP.HCM*”, “*T.U*”), точки и двоеточия в форме веб сайта (“*http://www.google.com*”) или дефис и другие знаки в форматах даты и времени (“2-9-2003”, “19:30”) и т. п;
2. Выделение фразы построено на анализе правого и левого окружения знаков препинания: конец предложения (фразы) фиксируется при наличии точки, двоеточия, точки с запятой, вопросительного или восклицательного знаков, многоточия и т. д. Знаки двоеточия и кавычек в предложениях с прямой речью или сообщении могут восприниматься как разделители фразы.
3. Распознавание собственных имен, сокращений, технических терминов, аббревиатур и иностранных лексем.

Для выявления в текстах графематических дескрипторов необходимо иметь правила структуры текстовых сегментов (образцы) и правила извлечения. Первые выявляют лингвистические свойства структуры текстов, тогда как вторые, используют эти свойства для распознавания текстовых фактов. Формирование таких правил в существующих разработках производится вручную, что является причиной сложности настройки системы графематического анализа.

Процесс извлечения опирается на сопоставление образцу, который задается при помощи правил на специализированном формальном языке. Правила определяют не только образец, но и действия, которые должны быть выполнены при успешном сопоставлении.

Модель извлечения должна оперировать с большим числом атомарных признаков, приписываемых фактам текста, и не должна привязываться к конкретному синтаксису. Для того, чтобы правила извлечения можно было использовать в качестве компонентов более сложных моделей, их синтаксис должен быть предельно простым. Недостатком многих разработок является сильная зависимость от конкретной грамматики языка. Ручное составление правил человеком-экспертом в большинстве случаев требует больших трудозатрат, кроме того, зачастую приводит к появлению правил противоречащих друг другу.

В данной статье предлагается и исследуется модель для извлечения графематических дескрипторов в системе обработки вьетнамского языка, основанная на символическом подходе с использованием методов, основанных на сопоставлении образцов (pattern-based) [3], которые оперируют понятиями "образцы" и правилами их сопоставления с фрагментами текстов. Также представлены и обсуждены лингвистические характеристики и атрибуты вьетнамского языка, понятия и конструкции образцов и правил для построения наборов правил извлечения фактов из текстов на вьетнамском языке. Ключевым элементом модели является набор правил извлечения. По экспериментальному начальному набору предложений, сгенерированному из 283 992 онлайн документов, построены набор правил извлечения фактов и новый набор предложений, которые могут использоваться в других приложениях.

Оснований текст статьи построен следующим образом. Во втором разделе излагается метод извлечения графематических дескрипторов. Модель и основные понятия описаны в третьем разделе. В четвертом — рассматривается графематическая модель вьетнамского языка. В пятом — описываются эксперименты. В заключении подводятся итоги.

2. Графематические дескрипторы

Образец — объект, который представляют собой шаблон фразы, состоящий из элементов, связанных отношением предшествования.

Каждый образец описывает структуру текста с определенным порядком элементов образца. При извлечении информации из некоторого текста на основе данного образца элементы текста должны следовать друг за другом в том же порядке, в каком следуют друг за другом соответствующие им элементы образца.

По структуре образец во многом схож с регулярным выражением и состоит из шаблона и атрибутов. Например, образец с имением "P-Date" описывает конструкции вида:

```
P-Date {/* ngày 12/07/1982 */
    P: word Day/Month/Year;
    C1: IsVietnameseWord(word) & word = "ngày";
    C2: Length(Day) = 2 & IsNumeric(Day);
    C3: Length(Month) = 2 & IsNumeric(Month);
    C4: Length(Year) = 4 & IsNumeric(Year);
};
```

Шаблон (секция с имением "P" — pattern) — это регулярное выражение, записанное относительно его *элементов*. В шаблоне образца "P-Date" конструкция состоит из четырех атомов: "word", "Day", "Month" и "Year".

Атрибуты каждого из атомов формулируются в критерии (секции "C_1", "C_2", "C_3", "C_4" — criterion). По сути, атрибуты — это функции с аргументами элементами шаблона "P". Например, в секции "C_1" указано что, атом "word" является вьетнамским словом (IsVietnameseWord(word)), и "word" — слово "*ngày*" (word = "*ngày*"). В секции "C_4" указано что, атом "Year" состоит из четырех символов (Length(Year) = 4), и "Year" - число (IsNumeric(Year)).

Графематический дескриптор — особый образец, состоящий из шаблона, атрибутов и дескриптора. Шаблона и атрибуты описывают структуру, а *дескриптор* — объект, который приписывается графематическому дескриптору и описывает его характеристики и свойства.

В графематическом дескрипторе кроме основной функции распознавания фрагмента текста при успешном сопоставлении шаблона включается также функция, которая приписывает этим фрагментам текста дескрипторы.

Графематические дескрипторы разбиты на классы (знаков пунктуации, цифровых комплексов, формул, собственных имен, сокращений и т. п.). Каждый класс графематического дескриптора описывает текст с определенной точки зрения. На основе множества

графематических дескрипторов можно построить аннотированный корпус текстов для удобно разрешения задач обработки текста.

Например, графематический дескриптор с имением “M-Date” описывает конструкции вида:

```
M-Date {/* 12/07/1982 [Date]*/
    M: “[Date]”;
    P: Day/Month/Year;
    C1: Length(Day) = 2 & IsNumeric(Day);
    C2: Length(Month) = 2 & IsNumeric(Month);
    C3: Length(Year) = 4 & IsNumeric(Year);
};
```

где дескриптор “[Date]” графематического дескриптора “M-Date” записан в секции с имением “M”, шаблон и атрибуты дескриптора записаны в секциях “P”, “C_1”, “C_2” и “C_3”.

Правило — основная единица языка. Правила представляются в виде “образец → действие”, где “образец” — образец для извлечения в узком контексте; “действие” — набор действий, выполняемых при успешном сопоставлении образца.

В задаче извлечения графематических дескрипторов основная цель — распознавание и приписывание дескрипторов. Действие правил состоит из распознавания фрагмента текста с шаблоном и приписывания ему дескриптора при успешном сопоставлении. Например, правило с имением “R-Date” описывает конструкции вида:

```
R-Date {/* ngày 12/07/1982 → ngày “[Date]” */
    O: P-Date;
    A: M-Date.P → M-Date.M;
};
```

В секции O — образец “P-Date” для сопоставления фрагментов текста. *Действие* (секция A — action) выполняется при успешном сопоставлении образца в секции O. Оно представляет собой графематический дескриптор “M-Date”. При успешном сопоставлении фрагмента текста и образца (“*ngày* Day/Month/Year”) с шаблоном графематического дескриптора “M-Data” (“Day/Month/Year”) фрагменту текста приписывается дескриптор “[Date]”.

Процесс распознавания и приписывания дескрипторов может иллюстрируются следующим примером. Рассмотрим предложение “*Tôi sinh ngày 12/07/1982*”. При извлечении с помощью правила

"R-Date", предложение сопоставляется с описанным выше образом правила "P-Date". Фрагмент текста "*ngày 12/07/1982*" успешно сопоставляется с образцом. Далее следует действие правила, по которому этот фрагмент текста будет сопоставляться с шаблоном графематического дескриптора "M-Date" ("Day/Month/Year"), и в результате распознается фрагмент "*12/07/1982*". После процесса приписывания дескриптора действия, мы получим итоговый результат — аннотированное предложение "*Tôi sinh ngày [12/07/1982][Date]*".

3. Модель извлечения графематических дескрипторов

Прежде всего, дадим определение самому понятию распознавания графематических дескрипторов с точки зрения теории формальных языков.

Пусть $\Sigma = \{\sigma_i\}$ — *алфавит* (конечное упорядоченное множество символов).

$\mathcal{L} \subseteq \Sigma^* = \{w = \langle\sigma_i\rangle | \sigma_i \in \Sigma, |w| \geq 0\}$ — *некоторый язык*, заданный над этим алфавитом.

Элемент текста — последовательность символов алфавитов языка: $w = \sigma_1\sigma_2 \ldots \sigma_l$, где l — длина элемента текста w.

Текстовый сегмент языка $\mathcal{L}$ представляет собой некоторую последовательность текстовых элементов вида: $s = w_1 w_2 \ldots w_n$, где s — некоторый текстовый сегмент, w_i — i-ый текстовый элемент сегмента (слог, числа или знак препинания). n - длина текстового сегмента s.

Под сцеплением сегментов $s_1 = w_1^1 \ldots w_l^1$ и $s_2 = w_1^2 \ldots w_k^2$ будем подразумевать такой сегмент $s = w_1^1 \ldots w_l^1 w_1^2 \ldots w_k^2$, что элементы с номерами $1 \ldots l$ совпадают с элементами сегмента s_1, а элементы с номерами $l+1 \ldots l+k$ совпадают с элементами сегмента s_2. Для отражения факта сцепления будем использовать запись

$$s = s_1 + s_2 + \ldots + s_n,$$

где s_i — текстовый сегмент и s — сцепление сегментов $s_1, s_2, \ldots, s_n$. Набор $p(s) = \langle s_1, s_2, \ldots, s_n\rangle$ называется разбиением сегмента s.

Набор текстовых сегментов $\mathcal{C}_\mathcal{L} = \{s_1, s_2, \ldots, s_N\}$ является конечной совокупностью текстовых сегментов.

Множества элементарных атрибутов — конечные совокупности $\mathcal{A} = \{A_i\}$, $A_j \neq \emptyset$, где $A_j \subseteq \mathcal{C}_\mathcal{L}$. Если текстовый сегмент $s \in \mathcal{C}_\mathcal{L}$

принадлежит A_j ($s \in A_j$), то будем считать, что s удовлетворяется атрибутом класса A_j или атрибут класса A_j покрывает сегмент s.

Образец или *набор элементарных атрибутов* — последовательность элементарных атрибутов вида: $P = \langle A_1, A_2, \ldots, A_k \rangle$, где k — длина набора, A_i (класс атрибута) — i-ый элемент набора. Для любой пары элементов образца (A_i, A_j) элемент A_i предшествует элементу A_j, если $i < j$.

Пусть $P = \langle A_1, \ldots, A_k \rangle$ — образец. Разбиение $p(s) = \langle s_1, s_2, \ldots, s_l \rangle$ сегмента s назовем *правым разбиением* по образцу P, если $l = k$ и $\forall j \in \overline{1,k} \quad s_j \in A_j$. Будем считать, что s покрывается образцом P или образец P покрывает сегмент s. Образец P задает множество таких текстовых сегментов S_P, что каждый сегмент $s \in S_P$ покрывается образцом P.

Под сцеплением образцов $P_1 = A_1^1 \ldots A_l^1$ и $P_2 = A_1^2 \ldots A_k^2$ будем подразумевать образец $P = A_1^1 \ldots A_l^1 A_1^2 \ldots A_k^2$. Для отражения факта сцепления будем использовать запись

$$P = \langle P_1, P_2, \ldots, P_n \rangle$$

где P_i — образец, и P — сцепление образцов $P_1, P_2, \ldots, P_n$.

Набор образцов $\mathcal{C}_\mathcal{A} = \{P = \langle A_i \rangle | A_i \in \mathcal{A}\}$ является конечной совокупностью образцов.

Дескриптор представляет собой запись, в которой описываются характеристики и свойства одного объекта.

Пусть задано некоторое *конечное множество дескрипторов*

$$\mathcal{T} = \{T_i | T_i - \textit{дескриптор}\}.$$

Графематический дескриптор — пара образца и дескриптора: $M_i = (P_i, T_i)$, где $P_i \in \mathcal{C}_\mathcal{A}$ — образец, и $T_i \in \mathcal{T}$ — дескриптор, в котором описываются характеристики и свойства образца "P_i".

Рассмотрим *конечное множество классов графематических дескрипторов*

$$\mathcal{C}_\mathcal{T} = \{M_i = (P_i, T_i)) | P_i \in \mathcal{C}_\mathcal{A} \wedge T_i \in \mathcal{T}\}, \ (|\mathcal{C}_\mathcal{T}| > 1),$$

где $P_i = \langle A_i \rangle \in \mathcal{C}_\mathcal{A}$ — характеризуется структурой графематического дескриптора, а T_i — дескриптор, в котором описываются характеристики и свойства графематического дескриптора. $M_0 \in \mathcal{C}_\mathcal{T}$ — *класс нераспознанных графематических дескрипторов*. Текстовый

сегмент $s \in \mathcal{C}_\mathcal{L}$ принадлежит классу M_i ($s \in M_i$), если существует правое разбиение сегмента s ($p(s) = \langle s_j \rangle$) по набору P_i. В этом случае будем считать, что s является графемой класса M_i. Кроме того, если строка s не является графемой с точки зрения правил языка, то будем считать по определению, что $s \in M_0$.

Определим *множество допустимых графем*:

$$\mathcal{L}_{\mathcal{C}_\mathcal{T}} = \{s \in \Sigma^* | \exists i > 0, s \in M_i\}.$$

Правила извлечения представляют собой выражения вида

$$R : (P_c, M_o = (P_o, T_o)) \to T_o,$$

где P_c — образец для извлечения в узком контексте, $M_o = (P_o, T_o)$ — графематический дескриптор, P_o — образец для извлечения фрагмента текста и T_o — дескриптор, который приписывается фрагменту. Правило говорит, что при успешном поиске в произвольном текстовом сегменте s фрагмента покрываемого образцом P_c (т. е. при существовании такого разбиения сегмента $p(s) = \langle s_1, \ldots, s_o, \ldots s_n \rangle$, в котором фрагмент текста s_o покрывается образцом P_o) текстовому сегменту s_o ставится в соответствие дескриптор T_o ($s_e \in M_o$).

Обозначим $\mathcal{R} = \{R\}$ — множество правил извлечения.

Пусть задан кортеж $\mathcal{M} = \langle \Sigma, \mathcal{C}_\mathcal{L}, \mathcal{C}_\mathcal{A}, \mathcal{C}_\mathcal{T}, \mathcal{R} \rangle$, где Σ — алфавит, $\mathcal{C}_\mathcal{L}$ — набор текстовых сегментов, $\mathcal{C}_\mathcal{A}$ — набор образцов, $\mathcal{T}$ — множество классов графем, $\mathcal{R}$ — множество правил извлечения. Основной *задачей распознавания графематических дескрипторов в условиях* $\mathcal{M}$ будем называть задачу построения для произвольного текстового сегмента $s \in \mathcal{L}$ ее набор классов графематических дескрипторов M_s.

4. Графематическая модель вьетнамского языка

4.1. Вьетнамские буквы

Во вьетнамском письме за основу взята латиница.
В алфавите 29 букв:
22 буквы английского алфавита

a, b, c, d, e, g, h, i, k, l, m, n, o, p, q, r, s, t, u, v, x, y;

без *f, j, w* и *z*, которые могут использоваться при написании иностранных названий и имен,

7 модифицированных букв с особыми диакритическими знаками:

đ, â, ă, ê, ô, ơ, ư.

Кроме того, *буквенные сочетания*: *ch, gh, gi, kh, ng, ngh, nh, ph, th, tr* — считаются отдельными буквами с их собственными разделами в словаре.

Вьетнамские гласные: a, â, ă, e, ê, i, o, ô, ơ, ư, y.

Вьетнамские согласные: b, c, d, đ, g, gh, gi, h, k, kh, l, m, n, ng, ngh, nh, ph, q, r, s, t, th, tr, v, x.

Вьетнамский язык является *тональным языком*. В орфографии, тон обозначается диакритическими знаками написанным выше или ниже гласной. Например, *à, á, ả, ã, ạ.*

4.2. Вьетнамские слоги

Слог во вьетнамском языке — это не только фонетическая единица, он почти всегда служит звуковой оболочкой морфологически значимой части слова или отдельного слова. В стандартном национальном орфографии есть 6 200 вьетнамских слогов. Слог может состоять из одного, двух, трех и четырех элементов. Состав слогов, содержащих от одного до четырех элементов, может быть представлен следующим образом:

(согласный + полугласный + гласный + согласный).

Присутствие третьего элемента слога обязательно; первый, второй или четвертый элемент слога могут отсутствовать.

Каждый слог во вьетнамском языке произносится определенным тоном, отличаясь от некоторых других слогов. Например, *“thanh”, “thành”, “thánh”.*

4.3. Вьетнамские слова

Вьетнамский язык является *изолирующим слоговым*. Среди вьетских языков он выделяется завершившимся процессом *моносиллабизации* (сокращением до одного слога исторически многосложных слов, отсутствием начальных стечений согласных) и тенденцией к

полисиллабизации (к образованию многосложных лексических единиц).

Вьетнамские слова могут состоять из одного или более слогов. Около 80% вьетнамских слов состоит из двух слогов (*“học sinh”, “quần áo”, “máy tính”*). Некоторые слова имеют три или четыре слога (*“xe gắn máy”, “trường đại học”, “máy tính xách tay”*).

Вьетнамский язык характеризуется отсутствием *словоизменения* и наличием *аналитических форм*. Слова во вьетнамском языке не изменяются по падежам.

4.4. Основные графематические атрибуты

Отметим основные атрибутные модели распознавания графематических дескрипторов:

Буквенные атрибуты

- *Вьетнамский слог* — произвольная последовательность букв вьетнамского алфавита, удовлетворяет некоторым правилам вьетнамского образования слога;
- *Иностранный слог* — произвольная последовательность Unicode-букв, которые не являются вьетнамскими слогами;
- *Слог* — объединение двух предыдущих классов;
- *Регистр* — определен на множестве слогов и принимает одно из следующих значений:
  - *нижний* — все символы слога находятся в нижнем регистре;
  - *заголовочный* — первый символ слова находится в верхнем регистре, остальные — в нижнем;
  - *верхний* — все символы слога находятся в верхнем регистре;
  - *смешанный* — любая другая комбинация регистров.
- *Место расположения* — атрибуты слога зависят от места расположения слога во фразе:

- *в начале фразы* — слог находится в начале предложения или после знаков разделителя фраз.
- *в конце фразы* — слог находится в конце предложения или перед знаками разделителя фраз.
- *в середине фразы* — слог находится между другими слогами в предложении.

Цифровые атрибуты

- *Число* — произвольная последовательность цифр;
- *Сложное Число* — произвольная последовательность цифр и символов точек, запятой, удовлетворяющая некоторым правилам образования чисел;
- *Буквенно-цифровая последовательность* — произвольная последовательность Unicode-букв и цифр:
  - *начальные цифры* — начальная часть последовательности — цифры, а последняя часть — буквы;
  - *начальные буквы* — начальная часть последовательности — буквы, а последняя часть — цифры;
  - *смешанный* — любая другая буквенно-цифровая последовательность.
- *Цифро-знаковый комплекс* — произвольная последовательность цифр и знаков:
  - *начальные цифры* — начальная часть последовательности — цифры, а последняя часть — знаки;
  - *начальные знаки* — начальная часть последовательности — знаки, а последняя часть — цифры;
  - *смешанный* — любая другая цифровоо-знаковая последовательность.
- *Сложный комплекс* — произвольная последовательность, состоящая из цифр, букв и знаков;

Знаковые атрибуты

- *Разделитель фразы* — произвольная последовательность символов точек (.), двоеточий (:), запятых (,), точек с запятой (;), восклицательных (!) или вопросительных знаков (?);
- *Открывающая скобка* — {, [, (, <, “.
- *Закрывающая скобка* — },],), >, ”.
- *Признак начала фразы* — произвольная последовательность открывающих скобок и знаков разделителя фразы.
- *Признак конца фразы* — произвольная последовательность закрывающих скобок и знаков разделителя фразы.
- *Признак начала параграфа* — присваивается первой графеме параграфа.
- *Формальные знаки* — знаки, которые используются в формуле математики, физики и т. п.

4.5. Графематические дескрипторы

Буквенные дескрипторы

- *Признак собственного имени* — присваивается графеме, если она квалифицирована как собственное имя (определен для множестве слов). Слово считается именем собственным, если оно содержится в специальном справочнике, либо находится в заголовочном регистре и не является первой графемой предложения.
  - *[NamePerson]* — дескриптор описывает имя человека;
  - *[NameNational]* — дескриптор описывает имя страны;
  - *[NameCity]* — дескриптор описывает название города;
  - *[NameOrg]* — дескриптор описывает название организации;
  - *[NameStreet]* — дескриптор описывает название дороги.

- *Аббревиатура* — сокращенное написание слова или группы слов, образованное из названий начальных букв или из начальных звуков слов, входящих в исходное словосочетание.
  - *[AbbrevPerson]* — дескриптор описывает аббревиатуру имени человека, например, “N. T. Trung”; “O. N. Granhichin”.
  - *[AbbrevOrg]* — дескриптор описывает аббревиатуру названия организации, например, “SmartFly, L.L.C.”.
- *Специальные слова и словосочетании*, которые используются на специальных предметах (математика, физика, информатика и т. д.)
  - *[WebAdd]* — комплекс, состоящий из букв, знаков и цифр, который удовлетворяет некоторым правилам образования формы Веб-адреса, например, “www.google.com”; “http://www.aot.ru/”.
  - *[Email]* — комплекс, состоящий из букв, знаков и цифр, который удовлетворяет некоторым правилам образования формы Email-адреса, например, “vkhhieukien@yahoo.com”; “hieukien@gmail.com.ru”.
  - *[Dir]* — комплекс, состоящий из букв, знаков и цифр, который удовлетворяет некоторым правилам образования формы директории, например, “C:|test.txt”.

Цифровые дескрипторы

- *[Num]* — цифро-знаковый комплекс, состоящий из цифр и знаков, который описывает числа, например, “12”; “1.000.000”; “15,500,000”.
- *[KeyNum]* — комплекс, состоящий из цифр, знаков и букв, который удовлетворяет некоторым особым правилам, например, “keynumber: LMRP-1200-1111-1774”; “car’s number: 52N-1008”.
- *[PhoneNum]* — цифро-знаковый комплекс, состоящий из цифр и знаков, который удовлетворяет некоторым правилам образования формы номера телефона, например, “054 820129”; “(04)84-873333”; “8905-289-1051”.

- *[Date]* — цифро-знаковый комплекс, состоящий из цифр и знаков, который удовлетворяет некоторым правилам образования формы даты (число, месяц, год), например, “12/07/1982”; “20-01-2001”.

- *[Time]* — комплекс, состоящий из букв, цифр и знаков, который удовлетворяет некоторым правилам образования формы времени, например, “12h”; “21h30ph”; “21:15:45”;“ 1’15”30”.

Знаковые дескрипторы

- *[MoP]* — дескриптор разделителя фразы, произвольная последовательность символов точек (.), двоеточий (:), запятых (,), точек с запятой (;), восклицательных (!) или вопросительных знаков (?);

- *[BoP]* — дескриптор начала фразы — произвольная последовательность открывающих скобок и знаков разделителя фразы.

- *[EoP]* — дескриптор конца фразы — произвольная последовательность закрывающих скобок и знаков разделителя фразы.

5. Реализация

Для эксперимента были взяты данные из 250 034 вьетнамских Интернет-документов, которые были получены с веб-сайта “http://www.tuoitre.com.vn/”. Для удобства и повышения эффективности статистической обработки данные были нормированы: исправлены форматы кода, отредактированы орфографические ошибки слогов. Тексты были сегментированы следующим образом: набор 100 статей был сохранен отдельным файлом, название которого присваивалось в соответствии с текущим порядковым номером. Затем с целью упрощения данных были удалены следующие части: автор текста, название текста, время создания текста, тип текста, тематика текста, источник. Удаление из текста некоторых объектов, не представляющих практического интереса для пользователей данных: рисунков, гиперссылок, схем, некоторых таблиц, формул и др. Вместо удаленного объекта вставлялся графематический дескриптор “[Object]” с указанием типа этого объекта. Начальные данные содержал 18 676 877 фраз и 131 318 974 слогов.

В целях исследования предложенного метода распознавания графематических дескрипторов, основанного на сопоставлении образцов, а также поиска путей эффективной реализации этого метода в программных средствах авторами была предпринята разработка модели графематического анализа текстов. В качестве технологической платформы для модели была использована платформа $C^{\#}$.

В модели атрибуты фрагментов текста описываются с использованием регулярных выражений. Например, атрибут "IsNumber(String s)" для проверки является ли фрагмент текста числом или нет, был задан регулярным выражением "$[0-9]^{+}([.][0-9]\{3\})^{*}$". При этом при успешном сопоставлении этого выражения получаются некоторые фрагменты текста: "100", "12.000.000", "25.545". Атрибут "IsFirstUpper(String s)" для проверки находится в верхнем регистре или нет первый символ фрагмента текста "s" был задан регулярным выражением "$[A-Z]([A-Za-z]^{*})$". В результате получаются допустимые фрагменты текста: "Hieu", "Saint Petersburg", "Viet Nam".

Структурные элементы текста, различные элементы текста, не являющиеся слогами (цифры и числа, даты в цифровых формах, буквенно-цифровые комплексы, цифро-знаковые комплексы и т. п.), аббревиатуры, фрагменты иностранного текста также размечаются автоматическим анализатором с высокой точностью.

Одним из важнейших результатов является разрешение проблемы распознавания собственных имен, которая традиционно представляет собой сложную задачу. Это связано с априорной невозможностью описания в словаре всего спектра возможных имен. Модель использует 21 контекстное правило извлечения для распознавания собственных имен. В том числе 117 090 разных имен людей, 258 имен стран, 4 707 названий городов и т. п.). В табл. 1 показан список слов, состоящий из имен людей, имен стран, названий городов.

Предложенный метод позволяет реализовать эффективный и быстродействующий модуль для распознавания графематических дескрипторов с высокой точностью. Точность распознавания графематических дескрипторов определяются путем случайного выбора некоторых распознаваемых фрагментов текста для проверки ошибок. После многократных проверок и вычислений средней ошибки, точность получившегося словаря составила 98%.

Таблица 1: Некоторые собственные имена: людей, стран и названий городов.

Имена людей	**Имена людей**	**Имена стран**	**Назв. городов**
Võ Nguyên Giáp	Vladimir Putin	Việt Nam	Saint Petersburg
Nguyễn Tấn Dũng	Barack Obama	Liên Bang Nga	Hà Nội
Võ Văn Kiệt	George Bush	Trung Quốc	Đà Nẵng
Nguyễn Minh Triết	Guus Hiddink	Hoa Kỳ	Budapest
Bảo Đại	Jose Mourinho	Hàn Quốc	Quảng Bình
Đặng Thùy Trâm	Alfred Riedl	Triều Tiên	Frankfurt
Đỗ Việt Khoa	George W. Bush	Nhật Bản	Sài Gòn
Nông Quốc Tuấn	Saddam Hussein	New Zealand	Bắc Sumatra
Trần Hồng Nam	Cristiano Ronaldo	Tây Ban Nha	Thượng Hải
Trịnh Công Sơn	Rio Ferdinand	Thụy Điển	Quảng Ninh

6. Заключение

В работе описана разработанная авторами модель извлечения графематических дескрипторов из текстов на вьетнамском языке, базирующаяся на специальном методе, основанном на сопоставлении образцов. Модель хорошо работает на практике и может извлекать имена людей, названия мест, организаций и еще многие графематические дескрипторы.

Разработанная модель может быть использована в различных задачах, связанных с обработкой неструктурированных и слабоструктурированных текстов. Возможными пользователями модели предполагаются не лингвисты, а эксперты в той или иной предметной области обрабатываемых документов, в чьи задачи входит описание лексики узкоспециализированных текстов (писем, объявлений и т. п.). Модель предоставляет довольно гибкие средства описания текстовых фактов в виде их шаблонов, которые используются для автоматического распознавания этих единиц в тексте, и позволяет записывать правила распознавания текстовых объектов и определения их атрибутов.

Список литературы

[1] *Граничин О.Н., Кияев В.И.* Информационные технологии в управлении. — М.: Изд-во Бином. 2008. 336 с.

[2] *Андреев А.М., Березкин Д.В., Симаков К.В.* Модель извлечения знаний из естественно-языковых текстов // Информационные технологии. 2007. № 12. С. 57–63.

[3] *Herve Dejean* Learning rules and their exceptions // Jounrnal of Machine Learning Research. V. 2. 2002.

[4] *Cao X.H.* Vietnamese — Some Questions on Phonetics, Syntax and Semantics. Nxb Giao duc — Hanoi. 2000.

[5] *Chu M.N., Nghieu V.P, Phien H.T.* Co so ngon ngu hoc va tieng Viet // Nxb Giao duc. — Hanoi, 1997. P. 142–152.

[6] *Кириллов В.И., Старченко А.А.* Логика. — М., 1995.

[7] *Седнов А.А.* Модель графематического анализа в системе обработки естественного языка // Системный анализ и информационные технологии. Вестник ВГУ — Изд-во Воронежского государственного ун-та. 2007. № 2. С. 69–77.

[8] *Андреев А.М., Березкин Д.В., Симаков К.В.* Метод обучения модели извлечения знаний из естественно-языковых текстов // Вестник МГТУ. Приборостроение. 2007. № 3. С. 75–94.

[9] *Андреев А.М., Березкин Д.В., Симаков К.В.* Модель извлечения фактов из естественно-языковых текстов и метод ее обучения // Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Труды восьмой всероссийской научной конференции (RCDL'2006) — Ярославль: Ярославский государственный университет. 2006. С. 252–262.

[10] *Александровский Д.А., Кормалев Д.А., Кормалева М.С., Куршев Е.П., Сулейманова Е.А., Трофимов И.В.* Развитие средств аналитической обработки текста в системе ИСИДА-Т // Тр. Десятой нац. конф. по искусственному интеллекту с междунар. участием КИИ-2006. Обнинск, 25-28 сентября 2006г.: ВЗт. — М.: Физматлит. 2006. Т. 2. С. 555–563.

[11] *Ле Ч. Х., Ле А. В., Ле Ч. К.* Автоматическое выделение слов и словосочетаний из вьетнамских печатных текстов // Стохастическая оптимизация в информатике. 2008. Вып. 4. С. 171-186.

[12] *Граничин О. Н., Хьеу Л. Ч.* Статистический способ выделения и словосочетаний из вьетнаиских печатных текстов // Вестник. СПбГУ: Изд-во С.-Петерб. ун-та. 2009. Серия 10 (Прикладная математика, информатика, процессы управления). Вып. 3. С. 161–169.

[13] *Le Trung Hieu, Le Anh Vu, Le Trung Kien* An unsupervised learning and statistical approach for Vietnamese word recognition and segmentation // Joining in The 2nd Asian Conference on Intelligent Information and Database Systems. Hue City. Viet Nam. March 2010.

[14] *Хьеу Л. Ч.* Обучение без учителя и статистический подход для сегментации и распознавания вьетнамских слов // Стохастическая оптимизация в информатике. 2009. Вып. 5. С. 193–208.