ИНФОРМАТИКА

2009 **январь-март** **№ 1**

УДК 004.912

С.Ф. Липницкий

АЛГОРИТМЫ ИНТЕРПРЕТАЦИИ СОДЕРЖАНИЯ ТЕКСТОВЫХ ДОКУМЕНТОВ НА ОСНОВЕ СИНТАКСИЧЕСКОГО ШАБЛОНА ПРЕДМЕТНОЙ ОБЛАСТИ

Предлагаются алгоритмы интерпретации содержания текстовых документов, основанные на шаблонах предложений в виде синтаксических деревьев с частично незаполненными вершинами (слотами). Алгоритмы могут быть использованы при перефразировании текстов с учетом требований к стилистической окраске, а также при составлении их аннотаций.

Введение

Согласно работе [1] под *интерпретацией* понимается «получение на основе одного исходного объекта (называемого интерпретируемым объектом) другого, отличного объекта, предлагаемого интерпретатором в качестве равносильного исходному на конкретном фоне ситуации, набора презумпций, знаний». Понятие интерпретации является базовым при построении интерпретирующей теории языка путем использования «информационного запаса» в виде знаний о предметной области. С учетом этой теории интерпретацию содержания реального текстового документа можно представить как «обогащение замысла» интерпретатора (компьютерной программы) за счет его «эрудиции», подкрепляемой базой знаний о предметной области в виде некоторой обобщенной синтаксической конструкции и специальных лексико-семантических средств (словарей).

В существующих программных продуктах задача интерпретации, или генерации (синтеза), связного текста решается главным образом путем использования лексических шаблонов предложений в виде готовых фрагментов текста [2, 3]. В процессе интерпретации нетерминальные переменные (пустые позиции) этих шаблонов заполняются конкретными фактографическими сведениями.

В данной статье для целей интерпретации (аннотирования) текстовых документов используются специальная база знаний, включающая обобщенное представление выходного текста в виде упорядоченного множества синтаксических шаблонов предложений, а также словари информативных словоформ и устойчивых словосочетаний. Предложенные алгоритмы интерпретации отличаются универсальностью, т. е. независимостью от предметной области. Настройка системы на конкретную предметную область реализуется путем формирования соответствующего синтаксического шаблона.

Задачу интерпретации содержания текстового документа будем решать в два этапа: на первом этапе сформируем кортеж синтаксических деревьев, используя синтаксический шаблон предметной области, а на втором синтезируем предложения текста.

1. Синтаксический шаблон предложения

Пусть имеется текст T в виде кортежа предложений. Обозначим через D_π синтаксическое дерево любого предложения π из текста T. Ордерево Dr_π, полученное из синтаксического дерева D_π заменой всех его поддеревьев, которые являются синтаксическими деревьями прагматически полных синтагматических структур (ПП-структур) [4], слотами («пустыми» вершинами), будем называть *синтаксическим шаблоном предложения* π (рис. 1). Под ПП-структурой понимается информативная в некотором тематическом разделе предметной области (т. е. хотя бы в одном тематическом корпусе текстов [4]) синтагматическая структура, выражаемая устойчивым словосочетанием (например, «информационные технологии», «входной язык», «радиоаппаратура»). Понятие синтагматической структуры есть обобщение понятия синтагмы на слу-

чай, когда ее членами являются не только слова, но и синтагмы. Синтаксические связи между словами синтагматической структуры представляются в виде ориентированного графа, в вершинах которого расположены вхождения слов в эту структуру, а дуги соответствуют синтаксическим связям между ними. Тематический корпус текстов – это некоторое непустое множество текстов входного языка по конкретной тематике, а объединение всех тематических корпусов называется полным корпусом текстов.

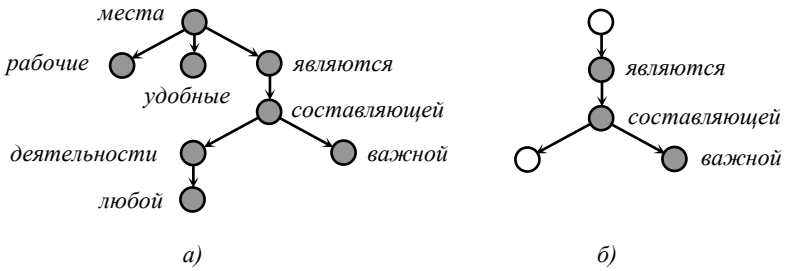

Рис. 1. Примеры синтаксического дерева и синтаксического шаблона предложения «Удобные рабочие места являются важной составляющей любой деятельности»: *а)* синтаксическое дерево; *б)* синтаксический шаблон

Синтаксические шаблоны предложений создаются в полуавтоматическом режиме: сначала на основе некоторых «хороших» текстов (например, рефератов текстовых документов) программно формируется совокупность таких шаблонов, а затем они корректируются экспертом-лингвистом. При формировании синтаксических шаблонов используется предварительно созданный словарь ПП-структур. Каждая ПП-структура включает хотя бы одно информативное (в каком-нибудь тексте) слово. Поэтому созданию словаря ПП-структур должно предшествовать формирование списка (словаря) информативных словоформ.

При синтезе синтаксического дерева предложения слоты заменяются синтаксическими деревьями синтагматических структур из семантического следа текста [4].

1.1. Алгоритм формирования словаря информативных словоформ

В отличие от информативности синтагматической структуры в некотором тексте [5] под информативностью словоформы a будем понимать условную вероятность того, что эта словоформа извлечена из тематического корпуса текстов (в котором она встречается с максимальной частотой) при условии, что она уже извлечена из полного корпуса текстов. На практике, как показано в статье [5], при достаточно больших объемах корпусов текстов информативность словоформы может быть вычислена по формуле

$$I_a = \frac{n_{\max}}{N},$$

где $n_{\max}$, N – абсолютные частоты словоформы a (с точностью до синонимии) в тематическом и полном корпусах текстов. При этом выбирается такой тематический корпус, в котором $n_{\max}$ принимает наибольшее значение по сравнению со всеми остальными корпусами.

Пусть $W_{\text{ПКТ}}$ – множество всех словоформ полного корпуса текстов. Обозначим через $W = \{W_a = \langle a, P_{\text{ПКТ}}, P_{\text{ТК-1}}, P_{\text{ТК-2}}, \ldots\rangle | a \in W_{\text{ПКТ}}\}$ *словарь словоформ*, где a – словоформа, $P_{\text{ПКТ}}$ и $P_{\text{ТК-}j}$ ($j = 1, 2, \ldots$) – ее абсолютные частоты соответственно в полном и j-м тематическом корпусах текстов. Пусть также $Inf = \{Inf_c = \langle c, I_c\rangle | c \in W_{\text{ПКТ}}\}$ – *словарь информативных словоформ*. Сформируем словарь Inf по следующему алгоритму.

А л г о р и т м 1. 1. На входе алгоритма – словарь словоформ W, на выходе – словарь информативных словоформ Inf. Алгоритм 1.1 включает следующие шаги:

1. $Inf := \varnothing$, $i := 1$.

2. Выбрать из словаря словоформ W кортеж $W_a = \langle a, P_{\text{ПКТ}}, P_{\text{ТК-1}}, P_{\text{ТК-2}}, \ldots\rangle$.

3. Найти в кортеже W_a максимальную частоту $P_{\text{ТК-}j}$, с которой словоформа a входит в тематический корпус текстов.

4. $I_a := P_{\text{ТК-}j} / P_{\text{ПКТ}}$.

5. Поместить кортеж $Inf_a = \langle a, I_a \rangle$ в словарь *Inf*.

6. Если элементы словаря *W* исчерпаны, то КОНЕЦ (словарь информативных словоформ сформирован). Иначе $i := i + 1$, перейти к п. 2.

1.2. Алгоритм формирования словаря ПП-структур

Процесс создания словаря ПП-структур реализуется в два этапа. На первом этапе программно формируется предварительный список словосочетаний. На втором этапе этот список просматривает эксперт-лингвист, удаляя из него некорректные словосочетания, в том числе словосочетания, синтаксическими графами которых не являются ордеревья (синтаксические деревья [5]). Алгоритм формирования предварительного списка ПП-структур работает следующим образом.

Пусть $\pi = a_1 a_2 \ldots a_i \ldots$ – произвольное предложение входного языка, а a_i – информативное слово этого предложения. В качестве параметров зададим пороговый уровень информативности слов $P_{\text{инф.}}$ [5] и максимально возможное число слов $N_{\max}$ в ПП-структуре. Последовательно присоединяя к слову a_i слева и справа другие слова предложения π, сформируем временный файл $F_{\text{врем.}}$ из полученных в результате двухсловных, трехсловных (и т. д.) подцепочек предложения π. Слово a_i также включим в файл $F_{\text{врем.}}$. Например, при $N_{\max} = 3$ для предложения π получим следующую последовательность словосочетаний: a_i; $a_{i-1}a_i$, $a_i a_{i+1}$; $a_{i-2}a_{i-1}a_i$, $a_{i-1}a_i a_{i+1}$, $a_i a_{i+1}a_{i+2}$. Аналогично поступим со всеми информативными словами и всеми предложениями полного корпуса текстов.

Используя полностью сформированный файл $F_{\text{врем.}}$, создадим множество Ch_0 всех однословных, двухсловных, трехсловных (и т. д.) подцепочек предложений полного корпуса текстов. Поставим в соответствие каждой из выбранных подцепочек α эмпирическую вероятность (относительную частоту) $P(\alpha)$ ее появления в полном корпусе текстов (с точностью до словоизменения и синонимии), т. е. фактически в файле $F_{\text{врем.}}$.

Зададим пороговое значение $P_{\text{порог.}}$ этой вероятности и удалим из множества Ch_0 все цепочки, вероятность появления которых в полном корпусе текстов меньше $P_{\text{порог.}}$. Обозначим через Ch_1 множество всех оставшихся в Ch_0 однословных цепочек, через Ch_2 – двухсловных цепочек и т. д. Обозначим, наконец, через Ch_j ($j \geq 1$) непустое множество из совокупности $\{Ch_1, Ch_2, \ldots\}$ с наибольшим индексом.

Удалим из множества Ch_{j-1} все цепочки, каждая из которых является подцепочкой некоторой цепочки из множества Ch_j. Аналогично поступим с цепочками множества Ch_{j-2}, являющимися подцепочками множества Ch_{j-1}, и т. д., заканчивая парой множеств Ch_1, Ch_2. Тогда все цепочки из множества $W_{\text{предв.}} = \bigcup_{i=1}^{j} Ch_i$ будем считать элементами предварительного списка ПП-структур. Опишем формально алгоритм реализации первого этапа создания словаря ПП-структур.

А л г о р и т м 1. 2. На входе алгоритма – полный корпус текстов *Fu*, словарь информативных словоформ *Inf*, на выходе – предварительный список словосочетаний $W_{\text{предв.}}$. Алгоритм 1.2 включает следующие шаги:

1. Эксперту-лингвисту установить значения порогового уровня информативности слов $P_{\text{инф.}}$, порогового значения $P_{\text{порог.}}$ вероятности $P(\alpha)$ и максимально возможного числа слов $N_{\max}$ в ПП-структуре.

2. $F_{\text{врем.}} := \varnothing$. $Ch_0 := \varnothing$, $Ch_1 := \varnothing$, $Ch_2 := \varnothing, \ldots, W_{\text{предв.}} := \varnothing$.

3. Выбрать из словаря *Inf* очередное информативное слово a_i.

4. Найти в полном корпусе текстов очередное предложение $\pi = a_1 a_2 \ldots a_i \ldots$, содержащее слово a_i.

5. Сформировать множество словосочетаний вида a_i; $a_{i-1}a_i$, $a_i a_{i+1}$; $a_{i-2}a_{i-1}a_i$, $a_{i-1}a_i a_{i+1}$, $a_i a_{i+1}a_{i+2}, \ldots$ (с учетом введенного экспертом-лингвистом максимально возможного числа слов $N_{\max}$ в ПП-структуре) и поместить эти словосочетания в файл $F_{\text{врем.}}$.

6. Если все предложения полного корпуса текстов, содержащие информативное слово a_i, найдены и обработаны, то перейти к п. 7, иначе – к п. 4.

7. Если все информативные слова из словаря *Inf* выбраны, то перейти к п. 8, иначе – к п. 3.

8. Поместить все цепочки файла $F_{\text{врем.}}$, состоящие из i слов, в множество Ch_i ($i = \overline{1, j}$).

9. $i := j$.

10. Удалить из множества Ch_{i-1} цепочки, каждая из которых является подцепочкой некоторой цепочки из множества Ch_i.

11. Если $i = 0$, то перейти к п. 12, иначе $i := i - 1$, перейти к п. 10.

12. $W_{\text{предв.}} := \bigcup_{i=1}^{j} Ch_i$.

На практике объем и качественный состав предварительного списка словосочетаний $W_{\text{предв.}}$ регулируется экспертом-лингвистом путем задания параметров $P_{\text{инф.}}$, $P_{\text{порог.}}$, $P(\alpha)$ и $N_{\max}$ (см. п. 1 алгоритма 1.2).

Обозначим множество ПП-структур, полученное в результате корректировки файла $W_{\text{предв.}}$ экспертом-лингвистом, через *Snt*.

2. Синтаксический шаблон предметной области

Некоторые вершины-слоты синтаксического шаблона предложения могут быть помечены, например, переменными [x], [y], Ω[x], Ω[y], Λ[x], Λ[y]. Будем называть такие вершины *поименованными слотами*. Назовем также метки слотов вида [x], [y], … *простыми*, а метки вида Ω[x], Ω[y], Λ[x], Λ[y] *составными*. (На практике поименованные слоты встречаются чаще всего в соседних синтаксических шаблонах.) Слоты упорядоченной совокупности синтаксических шаблонов, помеченные переменной вида [x], должны заполняться синтаксическим деревом одной и той же синтагматической структуры с точностью до предлогов и окончаний ее слов. Синтаксическому шаблону предложения со слотами вида Ω[x], Λ[y] должны предшествовать синтаксические шаблоны со слотами [x], [y]. Тогда слоты [x], [y] заполняются произвольными синтагматическими структурами, слот Ω[x] – родовой синтагматической структурой по отношению к структуре, которой заполнен слот [x], а Λ[y] – синтагматической структурой, синонимичной структуре, заполнившей слот [y] (Ω – отношение парадигматического подчинения, Λ – отношение синонимии [5]). Кортеж $Sh = \langle D_1, D_2, \dots, D_m \rangle$ организованных таким образом синтаксических шаблонов всех предложений текста $T = \langle \pi_1, \pi_2, \dots, \pi_m \rangle$ назовем *синтаксическим шаблоном* этого текста (рис. 2).

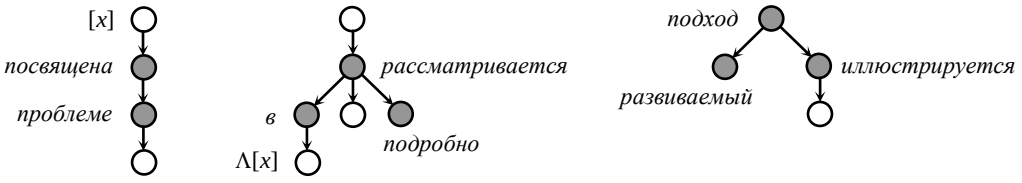

Рис. 2. Фрагмент синтаксического шаблона текста

2.1. Отношение замещения слотов

Процесс заполнения слотов синтаксического шаблона предложения формально регулируется отношением замещения слотов, которое определим следующим образом.

Рассмотрим некоторое непустое множество *Slo* поименованных слотов. Тогда бинарное отношение Ξ на этом множестве будем называть *отношением замещения слотов*, если для любых поименованных слотов $s_1, s_2 \in Slo$ соотношение $(s_1, s_2) \in \Xi$ выполняется тогда и только тогда, когда:

– поименованный слот s_1 является слотом вида [x];

– поименованный слот s_2 является слотом вида [x], Λ[x] или Ω[x];

– синтаксический шаблон предложения, содержащий слот s_1, предшествует синтаксическому шаблону, который включает слот s_2.

2.2. Отношение дискурсивной сочетаемости

Синтаксический шаблон текста строится на основе отношения замещения слотов Ξ и отношения дискурсивной сочетаемости Θ. Определим формально отношение Θ.

Пусть имеется множество $\{Sh_i \mid i = \overline{1, n}\}$ синтаксических шаблонов некоторых текстов. (На практике эти тексты и их синтаксические шаблоны формирует эксперт-лингвист.) Рассмотрим объединение множеств синтаксических шаблонов $Sh = \bigcup_{i=1}^{n} Sh_i$. Определим на множестве Sh антирефлексивное бинарное отношение Θ, такое, что для любых синтаксических шаблонов $D_r, D_s \in Sh$ некоторых предложений π_r и π_s соотношение $(D_r, D_s) \in \Theta$ справедливо тогда и только тогда, когда существует синтаксический шаблон текста Sh_j $(1 \le j \le n)$, элементами которого являются синтаксические шаблоны D_r и D_s предложений π_r и π_s соответственно, и в синтаксическом шаблоне текста Sh_j синтаксический шаблон предложения π_r непосредственно предшествует синтаксическому шаблону предложения π_s. Отношение Θ назовем *отношением дискурсивной сочетаемости* синтаксических шаблонов предложений.

Определим на множестве всех пар отношения Θ (т. е. на множестве Θ) отношение строгого порядка π (антирефлексивное и транзитивное бинарное отношение) следующим образом: будем считать, что для любых пар синтаксических шаблонов предложений (D_i, D_j) и (D_k, D_l) отношения Θ соотношение $(D_i, D_j)\ \pi\ (D_k, D_l)$ справедливо тогда и только тогда, когда $D_j = D_k$, т. е. D_j и D_k – один и тот же синтаксический шаблон.

Множество Sh с определенным на нем отношением дискурсивной сочетаемости Θ и строгим порядком π, заданным на множестве Θ, назовем *синтаксическим шаблоном предметной области*.

Используя строгий порядок π, синтаксический шаблон интерпретируемого текста можно построить в два этапа: сначала в виде ориентированного маршрута в графе, вершинами которого являются упорядоченные пары синтаксических шаблонов предложений, а дуги соответствуют отношению π (рис. 3, *а*), затем в виде орцепи, где вершины (синтаксические шаблоны предложений) соединены дугами, определяющими порядок использования этих шаблонов при синтезе текста (рис. 3, *б*). Выбор каждого очередного элемента множества Θ реализуется путем сравнения синтаксического шаблона и синтаксического дерева интерпретируемого предложения.

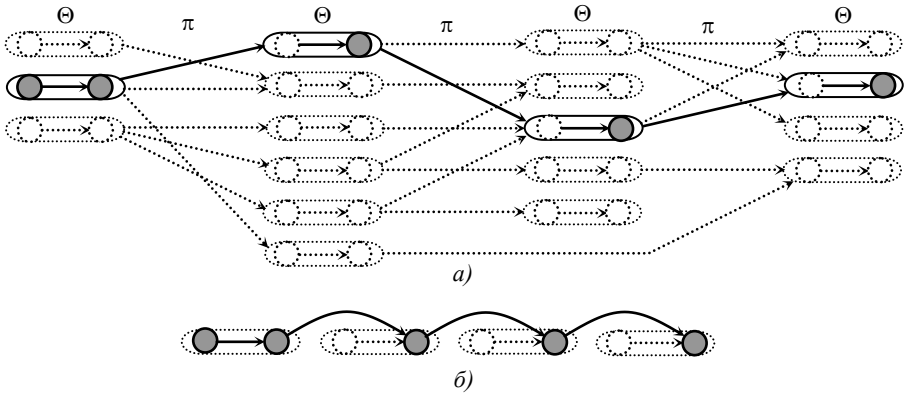

Рис. 3. Иллюстрация процесса построения синтаксического шаблона нового текста

На рис. 3, *а* изображен орграф отношения π, из которого исключены транзитивно замыкающие дуги, т. е. фактически изображен орграф редукции π′ $(\pi' = \pi \setminus \pi^2)$ отношения π.

Отношение дискурсивной сочетаемости строится для каждого вида синтезируемого текста, например аннотации, рецензии, доклада и пр., с учетом его стилистической окраски.

2.3. Алгоритмы синтеза кортежа синтаксических деревьев

Процедура построения кортежа синтаксических деревьев синтезируемого текста реализуется двумя алгоритмами.

В соответствии с первым алгоритмом ищется минимальный (в смысле строгого порядка π) элемент множества Θ, т. е. пара синтаксических шаблонов предложений (D_1, D_2). Шаблон D_1 должен удовлетворять следующему условию: в семантическом следе текста должны существовать ПП-структуры для заполнения слотов шаблона D_1. Далее в множестве Θ ищутся пары синтаксических шаблонов предложений (D_2, D_3), (D_3, D_4), … . Шаблоны предложений D_2, D_3, … также должны удовлетворять упомянутому выше условию. После заполнения слотов всех найденных шаблонов (кроме поименованных) ПП-структурами из семантического следа текста и ситуативно-синтагматической сети получим требуемый кортеж синтаксических деревьев с незаполненными поименованными слотами.

Второй алгоритм предназначен для заполнения поименованных слотов сформированного кортежа синтаксических деревьев.

Приведем более подробное описание этих алгоритмов, основываясь на системе словарей из [6] в качестве ситуативно-синтагматической сети.

А л г о р и т м 2. 1. На входе алгоритма – отношение замещения слотов Ξ, отношение дискурсивной сочетаемости Θ, семантический след текста $Ksd = \langle Sd_1, Sd_2, \ldots\rangle$, словарь ПП-структур $Snt = \{Snt_i = \langle \alpha, P_{ПКТ}, P_{TK\text{-}1}, P_{TK\text{-}2}, \ldots\rangle\}$, словарь синонимов $Sin = \{Sin_j = \langle \beta; \gamma_1, \gamma_2, \ldots\rangle\}$, парадигматический словарь $Par = \{Par_k = \langle \mu; \nu_1, \nu_2, \ldots\rangle\}$. (Здесь Sd_1, Sd_2, … – синтаксические деревья; α, β и μ – ПП-структуры; $P_{ПКТ}$ – частота встречаемости ПП-структуры α в полном корпусе текстов, а $P_{TK\text{-}i}$ – ее частота в i-м тематическом корпусе текстов; γ_1, γ_2, … – ПП-структуры, синонимичные структуре β; ν_1, ν_2, … – ПП-структуры, парадигматически подчиненные структуре μ.) На выходе алгоритма – кортеж синтаксических деревьев $Snt = \langle \Delta_1, \Delta_2, \ldots\rangle$ синтезируемого текста с незаполненными поименованными слотами. Алгоритм 2.1 включает следующие шаги:

1. $Snt := \varnothing$.

2. Используя словарь ПП-структур *Snt*, определить, какой из тематических корпусов текстов $TK \in \{TK\text{-}1, TK\text{-}2, \ldots\}$ является релевантным семантическому следу текста *Ksd*. Мощность пересечения множеств информативных синтагматических структур корпуса *TK* и семантического следа *Ksd* должна быть наибольшей.

3. Выбрать из множества Θ совокупность всех его минимальных элементов в смысле отношения π, т. е. множество упорядоченных пар синтаксических шаблонов предложений $Dh_1 = \{(D_1, D_2)_1, (D_1, D_2)_2, \ldots\}$.

4. Выбрать из множества Dh_1 такую упорядоченную пару $(D_1, D_2)_i$ и для каждого слота s_j (кроме поименованных) синтаксических шаблонов D_1 и D_2 найти в семантическом следе текста *Ksd* такое синтаксическое дерево синтагматической структуры Sd_j, которые удовлетворяют следующему условию: все орцепи длины 1 орграфов D_1 и D_2, в которых слот s_j заменен корнем ордерева Sd_j, являются ПП-структурами, т. е. содержатся в словаре *Snt*.

5. Заполнить каждый слот s_j синтаксических шаблонов D_1 и D_2 синтаксическим деревом Sd_j, т. е. образовать объединение орграфов D_1 (или D_2) и Sd_j, а вершину-слот s_j в полученном объединении заменить корнем ордерева Sd_j. Поместить полученные из шаблонов D_1 и D_2 синтаксические деревья Δ_1 и Δ_2 в кортеж *Snt*, сохранив метки всех поименованных слотов.

6. $k := 2$.

7. Выбрать из множества Θ совокупность всех упорядоченных пар синтаксических шаблонов предложений $Dh_k = \{(D_k, D_{k+1})_1, (D_k, D_{k+1})_2, \ldots\}$.

8. Выбрать из множества Dh_k такую упорядоченную пару $(D_k, D_{k+1})_i$ и для каждого слота s_j (кроме поименованных) синтаксического шаблона D_{k+1} найти в семантическом следе текста *Ksd* такое синтаксическое дерево синтагматической структуры Sd_j, которые удовлетворяют следующему условию: все орцепи длины 1 орграфа D_{k+1}, в которых слот s_j заменен корнем ордерева Sd_j, являются ПП-структурами.

9. Заполнить каждый слот s_j синтаксического шаблона D_{k+1} синтаксическим деревом Sd_j, т. е. образовать объединение орграфов D_{k+1} и Sd_j, а вершину-слот s_j в полученном объединении заменить корнем ордерева Sd_j. Поместить полученное из шаблона D_{k+1} синтаксическое дерево Δ_{k+1} в кортеж *Snt*, сохранив метки всех поименованных слотов.

10. Если выбранная в п. 8 упорядоченная пара $(D_k, D_{k+1})_i$ является максимальным элементом множества Θ, то КОНЕЦ, иначе $k := k+1$. Перейти к п. 7.

А л г о р и т м 2. 2. На входе алгоритма – семантический след текста $Ksd = \langle Sd_1, Sd_2, \ldots\rangle$, словарь синтагматических структур $Snt = \{Snt_i = \langle \alpha, P_{ПКТ}, P_{ТК\text{-}1}, P_{ТК\text{-}2}, \ldots\rangle\}$, словарь синонимов $Sin = \{Sin_j = \langle \beta; \gamma_1, \gamma_2, \ldots\rangle\}$, парадигматический словарь $Par = \{Par_k = \langle \mu; \nu_1, \nu_2, \ldots\rangle\}$ и кортеж синтаксических деревьев $\langle \Delta_1, \Delta_2, \ldots\rangle$ синтезируемого текста с незаполненными поименованными слотами. На выходе алгоритма – требуемый кортеж синтаксических деревьев $Knt = \langle D_1, D_2, \ldots\rangle$. Алгоритм 2.2 включает следующие шаги:

1. $k := 1$.

2. Если в синтаксическом дереве Δ_k отсутствуют поименованные незаполненные слоты с простыми метками, то перейти к п. 3, иначе – к п. 4.

3. $D_k := \Delta_k$. Поместить синтаксическое дерево D_k в кортеж *Knt*. Если все синтаксические деревья выбраны из кортежа $\langle \Delta_1, \Delta_2, \ldots\rangle$, то КОНЕЦ, иначе $k := k+1$. Перейти к п. 2.

4. Для каждого слота s_j синтаксического дерева Δ_k найти в семантическом следе текста *Ksd* такое синтаксическое дерево синтагматической структуры Sd_j, которое удовлетворяет следующему условию: все орцепи длины 1 орграфа Δ_k, в которых слот s_j заменен корнем ордерева Sd_j, являются ПП-структурами.

5. Заполнить каждый слот s_j синтаксического дерева Δ_k синтаксическим деревом Sd_j, т. е. образовать объединение орграфов Δ_k и Sd_j, а вершину-слот s_j в полученном объединении заменить корнем ордерева Sd_j. $D_k := \Delta_k$. Поместить синтаксическое дерево D_k в кортеж *Knt*.

6. Искать в синтаксических деревьях $\Delta_{k+1}, \Delta_{k+2}, \ldots$ слоты с составными метками, которые содержат простые метки слотов из синтаксического дерева Δ_k. Если составные метки найдены, то заполнить соответствующие слоты и перейти к п. 7, иначе перейти к п. 8.

7. $k := k+1$. Перейти к п. 2.

8. Исключить все метки вершин в синтаксических деревьях кортежа *Knt*. Заполнить их слоты (в случае их наличия). КОНЕЦ.

3. Алгоритмы интерпретации текста

3.1. Упорядоченное синтаксическое дерево

Определим предварительно понятие расстояния между словами предложения, отношение семантической близости и упорядочивающие отображения.

Расстоянием $R(a_i, a_j)$ между словами a_i и a_j цепочки $a_1a_2\ldots a_i\ldots a_j\ldots a_n$ назовем модуль разности j и i, т. е. $R(a_i, a_j) = |j - i|$.

Пусть a – произвольное слово некоторого предложения ($a \in V$, V – словарь), а L – множество синтаксически корректных синтагматических структур из полного корпуса текстов. (Факт синтаксической корректности устанавливает эксперт-лингвист.) Определим на множестве $\Omega \cap (\{a\} \times V)$ бинарное отношение $\succeq_a$, являющееся объединением эквивалентности $=_a$ и строгого порядка $>_a$, следующим образом. Будем считать, что для любых слов $b, c \in V$, таких, что $(a, b) \in \Omega$ и $(a, c) \in \Omega$, выполняется соотношение $(a, b) >_a (a, c)$, если в множестве L существует синтагматическая структура из слов a, b и c, такая, что $R(a, b) > R(a, c)$, и нет структуры из этих же слов, где выполняется неравенство противоположного знака. Считаем также, что $(a, b) =_a (a, c)$, если существует синтаксически корректная синтагматическая структура, где $R(a, b) = R(a, c)$ или найдутся две таких структуры, в которых соответственно $R(a, b) < R(a, c)$ и $R(a, b) > R(a, c)$. Отношение $\succeq_a$ назовем *отношением семантической близости*. Если $(a, b) >_a (a, c)$, то будем говорить, что слова a и b *семантически связаны сильнее*, чем слова

a и c. Если же $(a, b) =_a (a, c)$, то скажем, что слова b и c *семантически равнозначны относительно слова* a.

Для всех троек слов a, b, c типа рассмотренных выше построим совокупность отображений $\Phi_a : \Omega \cap (\{a\} \times V) \to \{1, 2, \ldots\}$, таких, что $\Phi_a((a, b)) > \Phi_a((a, c))$, если $(a, b) >_a (a, c)$, а если $(a, b) =_a (a, c)$, то $\Phi_a((a, b)) = \Phi_a((a, c))$. Такие отображения Φ_a назовем *упорядочивающими*.

На практике в качестве совокупности L используется полный корпус текстов, а образы всех дуг, исходящих из вершины a синтаксического дерева предложения, при упорядочивающем отображении Φ_a можно рассматривать как числовые метки на этих дугах.

Синтаксическое дерево любого предложения назовем *упорядоченным*, если все его дуги помечены натуральными числами, являющимися образами этих дуг при отображениях Φ_a (рис. 4).

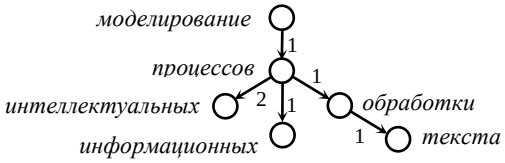

Рис. 4. Пример упорядоченного синтаксического дерева предложения
«Моделирование интеллектуальных информационных процессов обработки текста»

3.2. Алгоритм упорядочения синтаксических деревьев

А л г о р и т м 3.1. На входе алгоритма – совокупность отношений $\geq_a$, совокупность упорядочивающих отображений Φ_a и синтаксическое дерево некоторого предложения, на выходе – упорядоченное синтаксическое дерево. Алгоритм 3.1 включает следующие шаги:

1. Найти произвольную висячую вершину синтаксического дерева b_1, являющуюся конечной вершиной орцепи максимальной длины, и смежную ей вершину a.

2. Найти все дуги $(a, b_1), (a, b_2), \ldots$, исходящие из вершины a.

3. Найти натуральные числа, которые являются образами всех найденных в п. 2 дуг при отображениях Φ_a, и пометить ими эти дуги.

4. Условно исключить из синтаксического дерева все дуги, исходящие из вершины a, и их конечные вершины. Если в синтаксическом дереве после такого исключения имеются дуги, то перейти к п. 1, иначе – КОНЕЦ.

3.3. Алгоритмы синтеза текста

Рассмотрим в качестве примера процедуру синтеза предложения, упорядоченное синтаксическое дерево которого представлено на рис. 4.

На первом этапе синтеза строим синтагму «*Моделирование процессов*» (рис. 5).

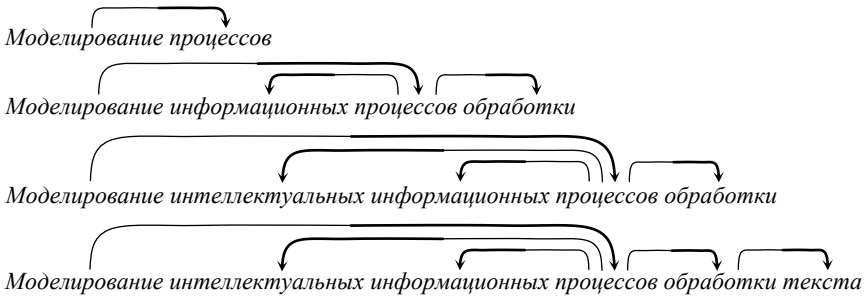

Рис. 5. Этапы синтеза предложения

На втором этапе строим синтагмы «*информационных процессов*» и «*процессов обработки*», поскольку их слова семантически связаны слабее, чем слова «*процессов*» и «*интеллектуальных*» (см. рис. 4).

На третьем этапе строится синтагма «*интеллектуальных процессов*».

На четвертом, последнем, этапе синтезируется синтагма «*обработки текста*». В результате получаем упомянутое выше предложение.

Прежде чем рассмотреть алгоритм синтеза синтаксически корректных предложений, введем следующее определение: произвольное конечное линейно упорядоченное множество A назовем *стеком*, если при исключении из него одного элемента исключается наименьший, а при добавлении – добавленный элемент становится наименьшим относительно заданного на A линейного порядка.

А л г о р и т м 3.2. На входе алгоритма – упорядоченное синтаксическое дерево D', содержащее только орцепи длины 1, на выходе – синтагматическая структура a. Алгоритм 3.2 включает следующие шаги:

1. $A := \varnothing$ (A – стек). Переменной x присвоить значение корня упорядоченного синтаксического дерева D', $a := x$, перейти к п. 2.

2. Упорядочить все вершины синтаксического дерева D' по неубыванию числовых меток, приписанных инцидентным им дугам, и поместить их в стек A в этом порядке, т. е. начиная с вершины, в которую заходит дуга с максимальной меткой.

3. Если $A := \varnothing$, то КОНЕЦ, иначе извлечь из стека A очередное слово и обозначить его через y.

4. Образовать синтагму xy (или yx) с использованием полного корпуса текстов для упорядочения слов x и y. Перейти к п. 3.

А л г о р и т м 3.3 (управляющий). Алгоритм включает следующие шаги:

1. Выполнить алгоритмы 2.1 и 2.2 синтеза кортежа синтаксических деревьев.

2. Выполнить алгоритм 2.1. Обозначить полученное в результате его работы упорядоченное синтаксическое дерево через D.

3. Искать поддерево D' ордерева D, включающего его корень и все орцепи длины 1, начальной вершиной которых является этот корень.

4. Выполнить алгоритм 3.1.

5. Исключить из орграфа D корень ордерева D' и все исходящие из него вершины.

6. Найти в любом компоненте связности орграфа D ордерево D', содержащее корень и все орцепи длины 1, начальной вершиной которых является этот корень. Если такое ордерево D' найдено, то исключить из орграфа D все дуги ордерева D', перейти к п. 4. Иначе – КОНЕЦ.

Заключение

Предложенные в статье алгоритмы интерпретации содержания текстовых документов могут быть использованы при перефразировании текстов с учетом требований к стилистической окраске, а также при составлении их аннотаций. Объем выходного текста определяется размером его семантического следа в ситуативно-синтагматической сети.

Благодаря использованию ситуативно-синтагматической сети возможен синтез аннотаций текстовых документов на различных выходных языках при наличии в системе соответствующих двуязычных словарей синтагматических структур.

Список литературы

1. Демьянков, В.З. Интерпретация, понимание и лингвистические аспекты их моделирования на ЭВМ / В.З. Демьянков. – М. : Изд-во Моск. ун-та, 1989. – 172 с.

2. Соколова, Е.Г. Автоматическая генерация текстов на ЕЯ (портрет направления) / Е.Г. Соколова, М.В. Болдасов // [Электронный ресурс]. – Режим доступа : http://www.dialog-21.ru/Archive/2004/Sokolova.htm. – Дата доступа : 24.06.2008.

3. Токарева, М.Ю. Автоматическая генерация спортивного комментария / М.Ю. Токарева, Е.И Большакова, Е.А. Бордаченкова // [Электронный ресурс]. – Режим доступа : http://www.dialog-21.ru/dialog2006/materials/html/tokareva.htm. – Дата доступа : 24.06.2008.

4. Липницкий, С.Ф. Модели знаний о предметной области для решения задач поиска и обработки текстовой информации / С.Ф. Липницкий // Информатика. – 2007. – № 2. – С. 25–34.

5. Липницкий, С.Ф. Математическая модель синтаксического анализа текста в информационно-аналитической системе / С.Ф. Липницкий // Информатика. – 2004. – № 1. – С. 28 – 36.

6. Липницкий, С.Ф. Семантический анализ текста на основе ситуативно-синтагматической сети / С.Ф. Липницкий // Информатика. – 2005. – № 2. – С. 102–110.

Поступила 22.10.08

Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail:lipn@newman.bas-net.by

S.F. Lipnitsky

ALGORITHMS FOR INTERPRETATION OF THE CONTENTS OF TEXT DOCUMENTS BASED ON THE SYNTACTICAL TEMPLATE OF APPLICATION DOMAIN

Interpretation algorithms of the contents of text documents based on sentence templates presented in the form of syntactical trees with partially empty tops (slots) are suggested. Algorithms can be used for rephrasing texts according to the requirements for stylistic coloring as well as for compiling summaries.