А. И. ЧУГУН, Т. А. НЕДЗЕЛЬСКАЯ

К ВОПРОСУ О ПОСТРОЕНИИ АЛГОРИТМА СИНТЕЗА ГЛАГОЛЬНЫХ ФОРМ РУССКОГО ЯЗЫКА

Процесс автоматической переработки текстовой информации содержит этап грамматической обработки отдельных словоформ на морфологическом уровне, т. е. определение морфологических характеристик (анализ) словоформ, синтез словоформ по их формальным и морфологическим признакам, классификация новых слов и т. п. вне зависимости от контекста. Автоматизация решения перечисленных задач предполагает применение для этой цели ЭВМ и, естественно, требует построения моделей, которые на формальном уровне отображали бы процесс грамматической обработки отдельных словоформ. В настоящее время построены и реализованы на ЭВМ действующие модели анализа некоторых глагольных форм и прилагательных русского языка [1, 2]. Цель данной работы — описание одного из возможных методов построения формальной модели синтеза. Предлагается алгоритм синтеза всего слова и отдельных словоформ на примере синтеза личных форм невозвратных глаголов русского языка.

В задачу синтеза включается восстановление отдельных словоформ и всей парадигмы того или иного слова по его метке, под которой понимается база парадигмы [3]. Восстановление отдельных словоформ по их меткам возможно при наличии самой метки и правила восстановления, поэтому блок-схему модели синтеза можно представить в виде последовательно соединенных двух блоков — формирующего базу (ФБ) и содержащего правила получения из нее словоформ, собственно синтезатора словоформ (СС) (рис. 1). Входной сигнал блока ФБ x_{i0} интерпретируем как одну из словоформ парадигмы слова x_i. Выходной сигнал этого

блока y_{iT_c} (он же входной сигнал для блока СС) — как базу парадигмы слова x_i. Выходной сигнал блока СС x_{ij} интерпретируем как новую словоформу из парадигмы слова x_i. Для упрощения блока ФБ ограничим множество словоформ, поступающих на его вход, множеством $A(x_{i0} \in A = \bigcup_{i=1}^{n} x_{i0})$.

Множество A конечно, и n ограничено словарем [4]. Эти ограничения позволяют нам использовать в качестве формирователя баз (ФБ) модель классификации глаголов по типам спряжения, описанную в работе [3]. На вход этой модели подают словоформы из множества A. На выходе получают базы парадигм соответствующих слов с указанием номера типа спряжения T_c, к которому относится то или иное слово. Блок СС содержит простое правило порождения словоформы x_{ij} — к базе подсоединить окончание — и само окончание для нее.

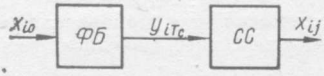

Рис. 1.

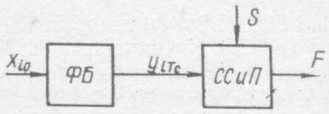

Рис. 2.

В пределах поставленной задачи мы будем рассматривать парадигмы, состоящие из личных спрягаемых форм невозвратных глаголов, причем только из их синтетических представителей. Из этого следует, что каждая парадигма будет состоять из 13 словоформ, $\bigcup_{j=1}^{13} x_{ij} = x_i$ $(j = 1, 2, 3, ..., 13)$ и одна из $x_{ij} = x_{i0}$. Отдельные словоформы в парадигме характеризуются определенными, только им присущими наборами грамматических категорий S. Для удобства закодируем эти наборы натуральными числами в следующем порядке: 1 — инфинитив; 2 — первое лицо, единственное число, непрошедшее время, изъявительное наклонение; 3 — второе лицо, единственное число, непрошедшее время, изъявительное наклонение; 4 — третье лицо, единственное число, непрошедшее время, изъявительное наклонение; 5 — первое лицо, множественное число, непрошедшее время, изъявительное наклонение; 6 — второе лицо, множественное число, непрошедшее время, изъявительное наклонение; 7 — третье лицо, непрошедшее время, изъявительное наклонение; 8 — единственное число, мужской род, прошедшее время, изъявительное наклонение; 9 — единственное число, женский род, прошедшее время, изъявительное наклонение; 10 — единственное число, средний род, прошедшее время, изъявительное наклонение; 11 — множественное число, прошедшее время, изъявительное наклонение; 12 — единственное число, второе лицо, повелительное наклонение;

13 — множественное число (единственное, форма вежливости), второе лицо, повелительное наклонение. При таком порядке кодирования

$$x_{i0} = x_{i1}$$

Восстановление слова по его метке требует порождения всех словоформ, входящих в его парадигму, а значит необходимо для каждого слова иметь 13 отдельных блоков СС. Все эти блоки включают в себя одно и то же правило порождения словоформ и отличаются только помещенными в них окончаниями, поэтому для восстановления слова лучше построить один блок, который будет содержать правило порождения и 13 окончаний парадигмы, расположенных в порядке возрастания номера кода. Назовем этот блок синтезатором словоформ и парадигмы слова (СС и П), так как его можно использовать для порождения как парадигмы, так и отдельных словоформ из нее.

Для этой цели на второй вход данного блока подается информация о коде словоформы. Таким образом, выходной сигнал блока СС и П будет зависеть от входного сигнала S — при $S = S_j$ на выходе получим x_{ij}, а при $S = 0$ на выходе будем иметь всю парадигму слова x_i. Элементы $x_{i0} \in A$, а соответственно и множество парадигмы, можно расклассифицировать по 124 типам спряжений, каждый из которых характеризуется своим набором окончаний. В соответствии с этим уменьшается и количество блоков СС и П до 124 (по количеству наборов окончаний).

В работе эти блоки собраны в один общий блок, который представлен в виде таблицы $T_c S_j$.

Наибольший интерес вызывает возможность порождения личных форм продуктивных глаголов, поэтому их блоки синтеза собраны в подблоке I таблицы и приведены полностью. Подблок II объединяет блоки синтеза 275 непродуктивных основ глаголов; ради экономии места он приведен частично. Блок-схему модели синтеза личных форм невозвратных глагалов можно представить в следующем виде (рис. 2). Эта модель универсальна и позволяет получить любое, без исключения, слово x_i или его отдельные словоформы x_{ij} по их представителям x_{i0} из открытого множества $B (x_{i0} \in B, A \subset B)$.

Модель реализует функцию F:

$$F(y, S) = \begin{cases} x_{ij}, & \text{если } S = S_j \\ x_i, & \text{если } S = 0, \end{cases}$$

где

$$y = y_i T_c (x_{i0}).$$

Алгоритм синтеза представляет собой набор формальных процедур, которые использовались при построении алгоритма классификации глаголов по типам спряжения, дополненный формаль-

Таблица T_c S_j

T_c \ S_j	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}	S_{13}	Номер подблока
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	ть	ю	ешь	ет	ем	ете	ют	л	ла	ло	ли	й	йте	
2	овать	ую	уешь	ует	уем	уете	уют	овал	овала	овало	овали	уй	уйте	
3	евать	юю	юешь	юет	юем	юете	юют	евал	евала	евало	евали	юй	юйте	
4	евать	ую	уешь	ует	уем	уете	уют	евал	евала	евало	евали	уй	уйте	
5	ять	ю	ешь	ет	ем	ете	ют	ял	яла	яло	яли	й	йте	
6	уть	у	ешь	ет	ем	ете	ут	ул	ула	уло	ули	и	ите	
7	уть	у	ешь	ет	ем	ете	ут	ул	ула	уло	ули	ь	ьте	
8	ить	ю	ишь	ит	им	ите	ят	ил	ила	ило	или	й	йте	
9	ить	ю	ишь	ит	им	ите	ят	ил	ила	ило	или	и	ите	
10	ить	ю	ишь	ит	им	ите	ят	ил	ила	ило	или	ь	ьте	
11	ить	лю	ишь	ит	им	ите	ят	ил	ила	ило	или	и	ите	
12	ить	лю	ишь	ит	им	ите	ят	ил	ила	ило	или	и	ьте	
13	дить	жу	дишь	дит	дим	дите	дят	дил	дила	дило	дили	ди	дите	I
14	дить	жу	дишь	дит	дим	дите	дят	дил	дила	дило	дили	дь	дьте	
15	зить	жу	зишь	зит	зим	зите	зят	зил	зила	зило	зили	зи	зите	
16	зить	жу	зишь	зит	зим	зите	зят	зил	зила	зило	зили	зь	зьте	
17	стить	щу	стишь	стит	стим	стите	стят	стил	стила	стило	стили	сти	стите	
18	ить	у	ишь	ит	им	ите	ят	ил	ила	ило	или	и	ите	
19	тить	чу	тишь	тит	тим	тите	тят	тил	тила	тило	тили	ти	тите	
20	тить	чу	тишь	тит	тим	тите	тят	тил	тила	тило	тили	ть	тьте	
21	сить	шу	сишь	сит	сим	сите	сят	сил	сила	сило	сили	си	сите	
22	сить	шу	сишь	сит	сим	сите	сят	сил	сила	сило	сили	сь	сьте	
23	ить	у	ишь	ит	им	ите	ат	ил	ила	ило	или	и	ите	
24	ить	у	ишь	ит	им	ите	ат	ил	ила	ило	или	ь	ьте	
25	ать	у	ешь	ет	ем	ете	ут	ал	ала	ало	али	и	ите	
...														
81	олоть	елю	елешь	елет	елем	елете	елют	олол	олола	ололо	ололи	ели	елите	II
82	оть	ю	ешь	ет	ем	ете	ют	ол	ола	оло	оли	и	ите	
...														
124	ять	ем-	емлешь	ем-	ем-	емлете	ем-	ял	яла	яло	яли	емли	емлите	

ными процедурами блока СС и П. В этот набор входят следующие процедуры: проверить окончание словоформ x_{io} на совмещение с эталонным окончанием; проверить букву или буквосочетание перед окончанием на совпадение с эталонными; проверить местоположение ударения в словоформе; сравнить x_{io}, если это необходимо, с эталонным словарем; в соответствии с T_c отбросить окончание и сформировать y_{iTc}; в соответствии с y_{iTc}; и S добавить требуемое окончание (окончания) и подать вновь полученную словоформу (словоформы) на выход. Работу алгоритма проиллюстрируем на примере восстановления словоформ глаголов, оканчивающихся в инфинитиве на — *ОТЬ*.

1. Сравнить окончание словоформы, поступающей на вход, с эталонным окончанием — *ОТЬ;* если ответ «да», то перейти к 3, если «нет» — к 2.

2. Сравнить окончания словоформ с другими эталонными окончаниями инфинитива (*-АТЬ, -ЯТЬ, -ЕТЬ, -УТЬ, -ЫТЬ, -ЗТЬ, -ИТЬ, -ЧЬ, -ТИ*).

3. Сравнить буквосочетание, стоящее перед окончанием на совпадение с *-ОЛ,* или *ОР-*; если «да», то перейти к 4, если «нет» — к 11.

4. Ударение падает на окончание словоформы (*ОТЬ*); если «да», то перейти к 5, если «нет» — к 11.

5. Сравнить словоформу со словарем-эталоном — *«МОЛОТЬ»,* если «да», то перейти к 6, если «нет» — к 9.

6. Присвоить $T_c=81$ и отбросить от словоформы — *ОЛОТЬ.*

7. Остаток словоформы подать на вход блока 81 таблицы T_cS_j (подблока II).

8. В соответствии с заданным S и полученным y_{iTc} добавить требуемое окончание (окончания) и вывести на печать новую словоформу (словоформы).

9. Присвоить $T_c=82$ и отбросить от словоформы —*ОТЬ.*

10. Остаток словоформы подать на вход блока 82 таблицы T_cS_j (подблок II) и перейти к 8.

11. Вывести на печать заданную словоформу с пометкой «Не глагол».

Предложенный метод построения модели синтеза отдельных словоформ по их меткам может быть использован при составлении других подобных моделей, а алгоритм синтеза слов и отдельных словоформ личных невозвратных глагалов — при построении системы автоматической обработки глаголов русского языка.

СПИСОК ЛИТЕРАТУРЫ

Соловьева Е. А. Автоматический морфологический анализ суженой парадигмы глагола. — В кн.: Проблемы бионики. Вып. 12, Харьков, 1974, с. 139—142.

Бондаренко М. Ф., Бузницкая Э. М. Алгоритм морфологического анализа имен прилагательных русского языка. — В кн.: Проблемы бионики. Вып. 12. Харьков, 1974, с. 149—156.