Ивашенко В.П. (Минск, БГУИР)

ОТ ТЕОРЕТИКО-МНОЖЕСТВЕННЫХ МОДЕЛЕЙ К СИМПЛИЦИАЛЬНЫМ МОДЕЛЯМ ЯЗЫКОВ

В информационных системах, основанных на знаниях, используются разные модели и языки представления знаний. Всё большее распространение получают системы, в которых знания представлены в виде семантических сетей или графовых информационных конструкций [1], рассматриваемых как модели текстов языков представления знаний. Преимущества систем, управляемых знаниями [2], определяются не столько наличием в них знаний об основной предметной области, сколько наличием знаний о способах их обработки и представления. Такие системы озадачены не только хранением знаний о предметной области, но и минимизацией издержек их хранения и обработки. Поэтому при разработке и работе систем подобного рода востребованы онтологии и модели, которые позволяют описать и сравнить используемые языки представления знаний с целью выбора более предпочтительных моделей или языков для представления знаний. В работе рассматривается подход, позволяющий единообразно описывать языки представления знаний, включая семантические сети.

Для описания и сравнения (формальных) языков, к которым относятся языки представления знаний, и фигур (геометрических форм) предложены теоретико-множественные модели языков [5], формальные грамматики (formal grammar) [6], [8], [9], [11], графовые грамматики (graph grammar) [10], [12], грамматики формы (shape grammar) [14].

Множества и строки. Традиционно тексты формальных языков называют строками или последовательностями и представляют в виде n-ок, скомпонованных из букв (символов) некоторого алфавита. Алфавит традиционно понимается как множество букв. Такие строки и множества могут рассматриваться как объекты разной природы или первые иногда сводят ко вторым. Чтобы исключить неоднозначность в вопросе о том как именно соотносятся строки и множества, рассмотрим варианты сведения строк ко множествам и остановимся на одном из них.

Рассмотрим множества S и T и бинарную операцию $\odot$ над ними:

$$S \odot T \underset{def}{=} S \cup \left\{\{T\} \cup \left\{\varnothing \cup \bigcup\nolimits_{N \in S} \left(\{N\} \cup \{\varnothing\}\right)\right\}\right\}. \qquad (1)$$

Последовательное применение этой операции к пустому множеству s и множествам T_i, где каждое одноэлементное множество T_i содержит букву a_i, приводит к следующим множествам:

$$\left(\left(S\odot T_1\right)\odot T_2\right)\odot T_3=\left(\left(\varnothing\odot T_1\right)\odot T_2\right)\odot T_3=\left(\left\{\left\{\varnothing,\left\{a_1\right\}\right\}\right\}\odot T_2\right)\odot T_3=$$

$$=\left\{\left\{\varnothing,\left\{a_1\right\}\right\},\left\{\left\{\varnothing,\left\{\varnothing,\left\{a_1\right\}\right\}\right\},\left\{a_2\right\}\right\}\right\}\odot T_3= \qquad (2)$$

$$=\left\{\left\{\varnothing,\left\{a_1\right\}\right\},\left\{\left\{\varnothing,\left\{\varnothing,\left\{a_1\right\}\right\}\right\},\left\{a_2\right\}\right\},\left\{\left\{\varnothing,\left\{\varnothing,\left\{a_1\right\}\right\},\left\{\left\{\varnothing,\left\{\varnothing,\left\{a_1\right\}\right\}\right\},\left\{a_2\right\}\right\}\right\},\left\{a_3\right\}\right\}\right\}$$

и т. д. Продолжая эту последовательность, можно получить представление строк произвольной конечной длины, начиная с пустой строки.

Следует отметить, что при таком представлении $S\subseteq S\odot T$, то есть каждая префиксная подстрока является подмножеством целой строки. Длина строки в этом случае может быть определена как мощность представляющего её множества $\left|S\right|=\left|\left(\left(\left(\varnothing\odot T_1\right)\odot T_2\right)\odot\ldots\right)\odot T_k\right|=k$.

Однако при этом, длина последовательной записи представления строк в виде таких множеств, зависит экспоненциально от длины представляемых строк, вызывая неудобства. Другой вариант сведения строк ко множествам избавлен от этого недостатка. Пусть для любого x

$$\left\{x\right\}_1\underset{def}{=}\left\{x\right\} \qquad (3)$$

и для любого натурального числа k $\left(\left\{\varnothing\right\}_k\right)$ верно

$$\left\{x\right\}_{k+1}\underset{def}{=}\left\{\left\{x\right\}_k\right\}, \qquad (4)$$

кроме того пусть для любого множества S выполняется равенство

$$1^S\underset{def}{=}S, \qquad (5)$$

примем также, что для любого натурального числа k и множества S выполняется

$$\left(k+1\right)^S\underset{def}{=}\left\{k^T\left|S\supseteq T\right.\right\}. \qquad (6)$$

Отметим, при k равном единице, в соответствии с последним выражением, 2^S есть степень (булеан) множества S.

Тогда строку можно представить множеством

$$\bigcup_{i=1}^{k}\left\{\left(k-i+1\right)^{\left\{a_i\right\}}\right\}_i, \qquad (7)$$

где k – длина строки, а a_i – i-я буква в строке.

В обоих вариантах представления длина строки равна мощности представляющего её множества $length\left(S\right)\underset{def}{=}\left|S\right|$.

Остановимся на втором варианте сведения строк ко множествам и условимся все строки в языках называть абстракциями, а строки длиной два и более – ассоциациями.

Отношения над строками. Пусть есть алфавит A. Множество всех строк длины n из букв алфавита A обозначается как декартова степень A^n алфавита A.

Операция конкатенации над строками определяется традиционным образом.

Множество всевозможных строк, получаемых в результате конечного числа применений операции конкатенации к пустой строке и элементам A^1, есть результат операции, подобной замыканию Клини, A^* над алфавитом A.

$$A^{*^1} \underset{def}{=} A^* \underset{def}{=} \bigcup_{n\in\mathbb{N}} A^n. \tag{8}$$

выразим ещё раз результат операции от A^*

$$A^{*^2} \underset{def}{=} \left(A^*\right)^* = \bigcup_{n\in\mathbb{N}} \left(A^*\right)^n = \bigcup_{m\in\mathbb{N}} \left(\bigcup_{n\in\mathbb{N}} A^n\right)^m. \tag{9}$$

Выражая замыкания получаемых множеств

$$A^{*^{n+1}} \underset{def}{=} \left(A^{*^n}\right)^*, \tag{10}$$

рассмотрим

$$A^{**} \underset{def}{=} \bigcup_{n\in\mathbb{N}} A^{*^n}. \tag{11}$$

В случае счётного алфавита A по построению полученное множество является счётным, т.е. $\left|A^{**}\right| = \left|A^*\right|$.

Таким образом, существует биекция между строками и подмножествами множеств A^{**} и A^*, а методы, применимые для исследования строк и подмножеств A^* применимы для исследования A^{**}.

Языком будем называть любое подмножество множества A^{**}. Тексты таких языков, являясь строками, имеют иерархическую структуру наподобие списков в языке LISP. Однако с классической математической точки зрения, в отличие от языка LISP, в общем случае структура текстов не является древовидной в силу того, что бинарные отношения равенства на множествах и строках являются антисимметричными. Отображение иерархической структуры текстов языков из A^* на A^{**} задаёт естественную теоретико-множественную интерпретацию, позволяющую исследовать структурные свойства языков и их текстов, используя классические математические свойства множеств. Каждый язык в A^{**} можно трактовать как отношение смешанной или определённой арности – множество кортежей на буквах из A или строках, а все отношения конечной арности на строках из A^{**} можно трактовать как язык из A^{**}. Таким образом, в A^{**} можно выделить язык подстрок, язык перестановок строк, язык конкатенаций строк, язык ассоциаций строк и другие. Перестановкой строки будем называть строку, множества компонентов которых совпадают, частным случаем перестановки строки является инверсия строки.

На множестве строк A^{**} определяются отношения, являющиеся мерами близости строк, позволяющие оценивать и вычислять аналоги редакционного расстояния. Для вычисления редакционного расстояния учитываются длины строк, необходимое количество удалений компонентов заданного типа (длины) из строки в заданную позицию, необходимое количество добавлений компонентов заданного типа в строку в заданную позицию, необходимое количество перестановок компонентов строки (заданного типа) в указанных

позициях. Причём это касается как самих исходных редактируемых строк, так и строк являющихся их компонентами. Редакционное расстояние вычисляется как значение функции $d(V)$ (частный случай – линейная комбинация), где V – набор вышеперечисленных характеристик.

Тексты рассматриваемых языков являются строками, в предположении того, что любой текст языка воспринимается в некотором порядке, соответствующему порядку элементов в строке, который, в частности, является порядком чтения этой строки некоторым последовательным алгоритмом.

Некоторые виды языков. Рассматриваемые языки по мощности могут быть конечными и бесконечными. Кроме этого языки могут быть ограничены по длине или по вложенности строк (горизонтально-ограниченные, вертикально-ограниченные). Для множества строк, ограниченных по длине значением p и ограниченных по вложенности значением q, примем обозначение $A^{(*\bmod p)^{*\bmod q}}$. Вычислительные машины и системы с конечным размером памяти (конечные автоматы) способны хранить только тексты конечных языков, ограниченные по длине и вложенности. Свойства операции конкатенации для таких языков подобны свойствам сложения по модулю для конечных групп.

В случае, когда алфавит является носителем линейного векторного пространства, например $\{0,1\}$ с операцией $\oplus$ векторного сложения по модулю два, строки одинаковой длины и одинаковой вложенности (типа) являются элементами линейного векторного пространства, а строки с линейно независимыми компонентами одинаковой длины и вложенности могут рассматриваться как симплексы.

Симплициальные языки – языки тексты, которых являются строками, каждый компонент которых – симплекс. Симплициальные языки (соответственно грамматикам формы) удобны для описания геометрических объектов и позволяют выстроить геометрическую модель языка. Тексты симплициальных языков могут обладать свойствами симплициальных комплексов и рассматриваться как алфавит более абстрактного языка (компоненты строк которого являются буквами этого алфавита).

Ассоциативным (ассоциативностным) языком будем называть язык, тексты которого являются строками, компоненты каждой из которых могут являться ассоциациями только других компонент той же строки. Хотя бы одна строка ассоциативного языка должна иметь ассоциацию. Язык, тексты которого не имеют ассоциаций, будем называть *диссоциативным*.

Графовые языки – ассоциативные языки, в текстах которых все ассоциации имеют длину два. Таким образом, семантические сети, как и SC-код [1], [3], соответствуют графовым ассоциативным языкам. В последнем случае sc-элементы (sc-терминалы, sc-узлы и sc-дуги) соответствуют абстракциям, а связки отношения их инцидентности – ассоциациям.

Каноническим языком будем называть язык, ни один текст которого не является перестановкой строки, неравной этой строке, которая является другим

текстом этого языка.

Симметричным языком будем называть язык, каждый текст которого, есть строка длины k, имеющая $k!$ текстов этого языка, которые являются перестановкой этой строки. Семантические сети, в том числе SC-код, могут быть представлены как в виде симметричного языка, так и в канонической форме – в виде канонического языка.

Известными классами языков являются генеративные языки, для каждого из которых существует порождающая грамматика, и негенеративные языки, для которых не существует таковой.

Отношения языков. Рассмотрим некоторые отношения на языках.

Подъязык – язык, являющийся подмножеством другого языка.

В соответствии с теоретико-множественными операциями можно выявить другие отношениями – отношения пересечения языков, отношение объединения языков, отношение разности языков (относительного дополнения), дополнение языка до A^{**}, декартово (прямое) произведение языков.

Ассоциацией языков будем называть подмножество декартова произведения двух или более языков. Обратное отношение – диссоциация.

Композиция языков – язык, полученный путём всех возможных конкатенаций строк первого языка со строками второго языка.

Вложение языка – строки одного языка получены путём изъятия из каждой из строк второго языка какой-либо подстроки.

Замыкание (ассоциативное) – вложение языка в минимальный *ассоциативный* либо *диссоциативный* язык.

Канонизация – отношение между неканоническим языком и его (максимальным) каноническим подъязыком.

Важным классом отношений является класс «вычислимых отношений» на языках, в частности – отношения алгоритмической трансляции текстов одного языка в тексты второго.

Описание генеративных языков в предлагаемом подходе возможно средствами, расширяющими существующие средства описания, например – РБНФ (расширенная форма Бэкуса-Наура) [13]. Для этого в алфавит (ассоциативно) расширенной формы Бэкуса-Наура вводится дополнительные символы $\langle\langle$ и $\rangle\rangle$, которые используются для указания начала и конца строки-компонента.

Предлагаемый подход позволяет исследовать свойства ассоциативных языков, в частности для языка и его подъязыка аналитически определить понятие ключевого элемента подъязыка – такого компонента текстов языка, который исключает свои ассоциации, меняет собственные и ассоциативные свойства ассоциированных с ним абстракций в текстах подъязыка по отношению к текстам языка [3].

Изложенный подход предлагается для моделирования текстов языков представления знаний, включая семантические сети, исследования и сравнению их свойств. В его рамках даны определения и перечислены некоторые виды языков и отношения между ними.

Результаты данной работы предназначены для использования при

аналитическом исследовании ассоциативных свойств языков, их сравнении, включая выявление ключевых элементов языка, и при применении результатов анализа свойств языков для оценки алгоритмической сложности обработки [4] сложных структур данных таких, как строки [7], списки, деревья и сети.

ЛИТЕРАТУРА

1. Голенков, В.В. Представление и обработка знаний в графодинамических ассоциативных машинах. — Минск, 2001.

2. Голенков, В.В., Гулякина, Н.А. Семантическая технология компонентного проектирования систем, управляемых знаниями. Материалы Международной научн.-техн. Конференции OSTIS, 2015. — Минск. — С. 57–78.

3. Ивашенко, В.П. Модели и алгоритмы интеграции знаний на основе однородных семантических сетей. Материалы Международной научн.-техн. Конференции OSTIS, 2015. — Минск. — С. 111–132.

4. Ивашенко, В.П., Татур М. М. Принципы платформенной независимости и платформенной реализации OSTIS. Материалы Международной научн.-техн. Конференции OSTIS, 2016. — Минск. — С. 145–150.

5. Маркус, С. Теоретико-множественные модели языков // Перевод с англ. М.В. Арапова; Под ред. Ю. А. Шрейдера. — Москва, 1970.

6. Пентус, А. Е., Пентус, М. Р. Теория формальных языков: Учебное пособие. — М., 2004.

7. Смит, Б. Методы и алгоритмы вычислений на строках: Пер. с англ. — М., 2006.

8. Calude, C., Marcus, S., Staiger, L. A topological characterization of random sequences,. Inform. Process. Lett. 88 (2003). — P. 245–250.

9. Chomsky, N. Three Models for the Description of Language. IRE Transactions on Information Theory. September, 1956.

10. Ehrenfeucht, A., Harju, T., Rozenberg, G. The Theory of 2-Structures: A Framework for Decomposition and Transformation of Graphs. World Scientific, 1999.

11. Handbook of Formal. Languages. / G. Rozenberg, Vol. 1, Word, Language, Grammar. Springer–Verlag, 1997.

12. Handbook of Graph Grammars and Computing by Graph Transformation Foundations. / G. Rozenberg, Vol. 1, World Scientific Publishing, Singapore, 1997.

13. ISO/IEC 14977:1996 — Information technology — Syntactic metalanguage — Extended BNF.

14. Stiny, G. (2006). Shape: Talking about Seeing and Doing. MIT Press, Cambridge, MA.