Алгоритм поиска решений уравнения на словах с одной переменной

Даниил Бидин
Алексей Угланов

В статье представлен алгоритм, применяемый для поиска решений уравнения на словах с одной переменной, что является одной из задач семиотики. Работа алгоритма продемонстрирована на примерах. Практическая реализация алгоритма выполнена на языке программирования Python 3.

Ключевые слова: алгоритм, уравнения на словах, семиотика, поиск, префикс, суффикс.

One-variable word equations: algorithm of solution

Daniil Bidin
Alexey Uglanov

In this article we would like to introduce the algorithm developed and designed to find one-variable word equation solutions, which is one of the semiotics problems. Algorithm's work is demonstrated by examples. Practical implementation has been made by means of Python 3 programming language.

Key-words: algorithm, word equations, semiotics, search, prefix, suffix

Введение

Проблема поиска заданного слова в тексте является основой любого поискового алгоритма. Первые алгоритмы поиска появились в 60-70 годах прошлого века, и с тех пор их применение и усовершенствование не теряет своей актуальности. В настоящее время научному обществу требуются так называемые «быстрые» алгоритмы, т.е. алгоритмы с маленькой сложностью (увеличение объёмов информации для обработки не влечёт за собой экспоненциальный рост времени работы программы). Эта необходимость связана с проблемой поиска информации в условиях многократного увеличения мировых объёмов информации, отсутствии структурированности около 80% из них, а также информационным шумом. Одной из областей науки, требующей подобный алгоритм, является задача по решению уравнения на словах.

Эта задача является двойственной следующей задаче: «является ли некоторое подмножество слов кодом?» Опишем алгоритм решения уравнения на словах с одной переменной и продемонстрируем его работу.

Вводные определения и постановка задачи

Пусть $\mathbf{\Pi} = \{\boldsymbol{a_1}, \boldsymbol{a_2}, \dots, \boldsymbol{a_n}\}$ – конечный алфавит постоянных, т.е. букв. <u>Длиной</u> слова называется количество букв (символов) в нем. Обозначим длину слова $\boldsymbol{Y}$ как $|\boldsymbol{Y}|$. Слово, в котором нет ни одного символа, называется <u>пустым</u>, оно имеет длину, равную $\mathbf{0}$, и обозначается $\boldsymbol{\lambda}$. Два слова называют <u>равными в лексикографическом смысле</u> тогда и только тогда, когда они состоят из одинаковых букв, записанных в одном и том же порядке. Это обозначается как $\boldsymbol{A} \equiv \boldsymbol{A}$. <u>Конкатенацией</u> $\boldsymbol{AB}$ (операцией приписывания) слов $\boldsymbol{A}$ и $\boldsymbol{B}$ называется слово, полученное в результате приписывания первого слова ко второму. Слово $\boldsymbol{A}$ называется <u>подсловом</u> слова $\boldsymbol{D}$ тогда и только тогда, когда существуют такие слова $\boldsymbol{B}$ и $\boldsymbol{C}$, что конкатенация $\boldsymbol{BAC} \equiv \boldsymbol{D}$. <u>Степенью</u> слова $\boldsymbol{C}$ называется слово $\boldsymbol{D}$ вида $\boldsymbol{C^n}$. Слово $\boldsymbol{C}$, в свою очередь, является <u>корнем</u> слова $\boldsymbol{D}$, если существует число $\boldsymbol{n}$ такое, что $\boldsymbol{C^n} \equiv \boldsymbol{D}$. <u>Кодом</u> называется множество слов $\mathbf{B} = \{\boldsymbol{b_1}, \boldsymbol{b_2}, \dots, \boldsymbol{b_n}\}$ в алфавите $\mathbf{\Pi} = \{\boldsymbol{a_1}, \boldsymbol{a_2}, \dots, \boldsymbol{a_n}\}$, если не существует такой перестановки символов $\{\boldsymbol{i_1}, \boldsymbol{i_2}, \dots, \boldsymbol{i_k}\}$, что $\boldsymbol{b_{i1}}, \boldsymbol{b_{i2}}, \dots, \boldsymbol{b_{ik}} \equiv \boldsymbol{b_{j1}}, \boldsymbol{b_{j2}}, \dots, \boldsymbol{b_{je}}$.

<u>Уравнение на словах</u> – совокупность букв алфавита и неизвестных переменных, являющихся словами, составленными из букв данного алфавита. <u>Решением</u> уравнения является такое слово, которое при подстановке в исходное уравнение даёт лексикографическое равенство двух частей уравнения. <u>Общим решением</u> уравнения называется множество всех решений уравнения.

Алгоритм позволяет найти множество общих префиксов и суффиксов слова $\boldsymbol{UV}$, возникающего при конкатенации префикса $\boldsymbol{U}$ и суффикса $\boldsymbol{V}$ слов $\boldsymbol{L}$ или $\boldsymbol{R}$ ($\boldsymbol{L}$ и $\boldsymbol{R}$ – соответственно левая и правая части данного уравнения; условия выбора $\boldsymbol{U}$ и $\boldsymbol{V}$ указаны ниже).

Результат поиска входит в формулу

$$\boldsymbol{X = (Z_1 Z_2)^k Z_1} \qquad \textbf{(1)}$$

являющуюся универсальной формулой решения уравнений на словах с одним неизвестным (формула выведена Ю.И. Хмелевским, им же доказано существование описываемого нами алгоритма **[1]**; доказательство сложности алгоритма и поиск общего решения были выполнены в работе М.Н. Максименко **[2]**) при следующих условиях:

1. одинаковое количество вхождений неизвестного в левую и правую части данного уравнения на словах (далее – д.у.);
2. нетривиальность д.у. (левая и правая части д.у. – не пустые слова);
3. несократимость д.у. (первые буквы левой и правой части д.у. не совпадают);

Уравнения на словах несократимого типа могут быть представлены в двух видах:

$$\boldsymbol{u_1 x u_2 x \dots u_n x \equiv x v_1 x v_2 \dots x v_m} \qquad \textbf{(2)}$$

$$\boldsymbol{u_1 x u_2 x \dots u_n x u_n \equiv x v_1 x v_2 \dots x v_m} \qquad \textbf{(3)}$$

4. непротиворечивость д.у. (первые или последние буквы левой и правой части совпадают и не принадлежат алфавиту **П**).

Слово $\boldsymbol{U}$ является префиксом $\boldsymbol{L}$ или $\boldsymbol{R}$, заканчивающимся до первого вхождения неизвестного $\boldsymbol{X}$ в соответствующую часть д.у. Слово $\boldsymbol{V}$ в случае ($\mathbf{2}$) является суффиксом $\boldsymbol{R}$ или $\boldsymbol{L}$, начинающимся после последнего вхождения неизвестного $\boldsymbol{X}$ в соответствующую часть д.у. В случае ($\mathbf{3}$) слово $\boldsymbol{V}$ является суффиксом $\boldsymbol{L}$ или $\boldsymbol{R}$.

Алгоритм применяется к строке, являющейся конкатенацией $\boldsymbol{UV}$. До конкатенации каждое из слов сокращается до своего минимального по длине корня, если такой имеется. $\boldsymbol{Z_1}$ и $\boldsymbol{Z_2}$ – непересекающиеся суффикс и префикс $\boldsymbol{UV}$, при которых слова $\boldsymbol{U}$ и $\boldsymbol{V}$ удовлетворяют условию сопряженности ($\boldsymbol{UV \equiv VU}$).

В своей работе М.Н. Максименко предполагает, что для поиска $\boldsymbol{Z_1}$ и $\boldsymbol{Z_2}$ необходимо использовать алгоритм Кнута-Морриса-Пратта (далее – КМП) для поиска подстроки в строке **[3]**. При попытке воспользоваться алгоритмом КМП мы пришли к выводу, что использование его классической реализации нецелесообразно для данной прикладной задачи. Алгоритм, предложенный нами, сравнивает первые $\boldsymbol{n}$ символов (префикс) $\boldsymbol{UV}$ с последними $\boldsymbol{n}$ символами (суффикс) $\boldsymbol{UV}$ для целых значений $\boldsymbol{n}$ от $\mathbf{1}$ до длины слова $\boldsymbol{UV}$ за вычетом единицы, т.к. само слово $\boldsymbol{UV}$ не должно входить в множество решений.

Получившееся в результате обработки алгоритма множество значений является множеством предполагаемых значений $\boldsymbol{Z_1}$ и $\boldsymbol{Z_2}$. Необходима проверка каждого предполагаемого значения не только на соответствие вышеуказанному условию, но и на получение лексикографического равенства при подстановке конечной формулы ($\mathbf{1}$) в диофантово уравнение. В нашем случае формулу необходимо проверить для множества целых значений $\boldsymbol{k}$ от $\mathbf{0}$ до $\mathbf{2}$ включительно, т.к. дальнейшие действия нецелесообразны **[1]**. Значения $\boldsymbol{X}$, удовлетворившие всем вышеперечисленным условиям, являются решениями данного уравнения на словах.

Работа алгоритма на примере уравнения на словах

Дано уравнение $\boldsymbol{aabbbxbbaabbbx} = \boldsymbol{xbbxbbaabbbaab}$, удовлетворяющее условиям нетривиальности, несократимости, непротиворечивости.

1. Левая часть $\boldsymbol{L} = \boldsymbol{aabbbxbbaabbbx}$, правая часть $\boldsymbol{R} = \boldsymbol{xbbxbbaabbbaab}$.
2. Слово $\boldsymbol{U} = \boldsymbol{aabbb}$ – префикс слова $\boldsymbol{L}$ до первого вхождения $\boldsymbol{X}$; а слово $\boldsymbol{V} = \boldsymbol{bbaabbbaab}$ – суффикс слова $\boldsymbol{R}$ после последнего вхождения $\boldsymbol{X}$.
3. Слово $\boldsymbol{V}$ – это вторая степень слова $\boldsymbol{bbaab}$, т.е., $\boldsymbol{bbaab}$ – корень $\boldsymbol{V}$, следовательно, присвоим $\boldsymbol{V} = \boldsymbol{bbaab}$.
4. Конкатенация $\boldsymbol{UV} = \boldsymbol{aabbbbbaab}$. Найдём множество всех префиксов этого слова, одновременно являющихся его же суффиксами. После всех итераций получим множество предполагаемых значений $\boldsymbol{Z_1}$ или $\boldsymbol{Z_2}$, равное $\{\boldsymbol{aab}\}$.
5. Рассмотрим слово $\boldsymbol{U}$. За значение $\boldsymbol{Z_2} = \boldsymbol{bb}$ примем оставшийся после подстановки префикса $\boldsymbol{Z_2} = \boldsymbol{aab}$ суффикс.
6. Рассмотрим слово $\boldsymbol{V}$. Так как конкатенация $\boldsymbol{Z_1 Z_2} \equiv \boldsymbol{V}$, значит, данные значения $\boldsymbol{Z_1}$ и $\boldsymbol{Z_2}$ можно проверять при подстановке в формулу ($\mathbf{1}$): $\boldsymbol{X} = (\boldsymbol{Z_1 Z_2})^{\boldsymbol{k}} \boldsymbol{Z_1}$.
7. При $\boldsymbol{k} = \mathbf{0}$ после подстановки $\boldsymbol{X} = \boldsymbol{aab}$ в начальное уравнение, получим лексикографическое равенство двух частей, следовательно, $\boldsymbol{X_1} = \boldsymbol{aab}$ является решением. Аналогично для $\boldsymbol{k} = \mathbf{1}$ и $\boldsymbol{k} = \mathbf{2}$ получаем $\boldsymbol{X_2} = \boldsymbol{aabbbaab}$ и $\boldsymbol{X_3} = \boldsymbol{aabbbaabbbaab}$.
8. Итоговые значения $\boldsymbol{X_1}, \boldsymbol{X_2}, \boldsymbol{X_3}$ – частные решения данного уравнения на словах, что и требовалось найти.

Заключение

Если уравнение на словах с одной переменной имеет решение, то объединение алфавитов $\boldsymbol{\Pi_1} \cup \{\boldsymbol{X}\}$ не является кодом, где $\boldsymbol{\Pi_1}$ – алфавит, состоящий из подслов левой и правой частей д.у. между вхождениями $\boldsymbol{X}$.

В дальнейшем мы планируем рассмотреть проблему поиска решений усложнённых уравнений (введение новых переменных или усложнение структуры существующих уравнений) и поиска зависимостей данных уравнений с диофантовыми уравнениями и их решениями.

Список литературы

1. Хмелевский, Ю.И. "Уравнения в свободной полугруппе" / Ю.И. Хмелевский // Тр. МИАН СССР, 107 – М., 1971. – С.3-288.
2. Максименко М. Н. "Общее решение уравнения на словах с одной переменной" / М. Н. Максименко // Научные Труды Вольного экономического общества России: 4-я Междунар. науч. - практ. конф. им. А. И. Китова "Математические методы и информационные технологии в экономике и управлении" РЭУ им. Г. В. Плеханова 27-28 марта 2014 г., 186 / Вольное экон. о-во России, Рос. экон. ун-т им. Г.В. Плеханова. – М., 2014. – С. 88-93.
3. Максименко М. Н. "О коротких решениях уравнения на словах с одной переменной" // Труды Междунар. науч. конф. 28/IX – 2/X 2015г. "Образование, наука и экономика в ВУЗах и школах. Интеграция в международное образовательное пространство", 1 – Горис, Армения, 2015.

Код УДК

519.765 Математическая лингвистика. Общие вопросы. Математические исследования языков, имеющие общий характер.

Приложение 1. Практическое применение. Реализация на языке программирования Python 3.6

- https://github.com/oogl/0004_reu_discrete_maths_word_equations
  или
- https://ideone.com/TaIPQ7