

СОПОСТАВЛЕНИЕ ТРАСС С ШАБЛОНАМИ В ЗАДАЧАХ АНАЛИЗА ПОВЕДЕНИЯ ПРОГРАММ

Хашаев Артур Акрамович

Студент

Факультет ВМК МГУ имени М. В. Ломоносова, Москва, Россия

E-mail: arthur@khashaev.ru

Одним из основных направлений интеллектуального анализа данных является выявление шаблонов (*pattern mining*), которое находит применение в таких областях, как биоинформатика и поиск недостатков в программах [1]. Перспективным направлением является применение данного метода в задаче обнаружения аномальных действий пользователей для обнаружения атак на веб-приложения. Однако на практике необходимы эффективные подходы к сопоставлению трасс событий с уже выявленными шаблонами, в том числе пригодные для использования в системах реального времени.

Определение 1. Назовем $\pi = (P, \preceq, \lambda)$ шаблоном, если $(P, \preceq)$ — конечное частично упорядоченное множество, $\lambda: P \rightarrow \Sigma$ — функция разметки.

Алфавит Σ можно интерпретировать как множество возможных действий пользователя или событий системы, а строки $\gamma \in \Sigma^*$ — как возможные трассы.

Определение 2. Множеством линеаризаций шаблона π назовем $\mathcal{L}(\pi) = \{\lambda(p_1) \cdots \lambda(p_n) : [p_1, \dots, p_n] \in \mathcal{L}(\preceq)\}$, где $\mathcal{L}(\preceq)$ — множество всевозможных линейных расширений частичного порядка $\preceq$.

В настоящей работе рассматриваются следующие задачи.

Задача 1 ($\alpha \models \pi$). Проверить сопоставляемость строки $\alpha \in \Sigma^*$ с шаблоном π — существование подпоследовательности $\gamma \in \mathcal{L}(\pi) : \gamma \sqsubseteq \alpha$.

Данная проверка позволяет обнаруживать точные совпадения с заданными шаблонами в потоке событий.

Задача 2. Найти $\text{LCS}(\alpha, \pi) = \max \{\text{LCS}(\alpha, \beta) : \beta \in \mathcal{L}(\pi)\}$, где $\text{LCS}(\alpha, \beta) = \max \{|\gamma| : \gamma \sqsubseteq \alpha \wedge \gamma \sqsubseteq \beta\}$ — длину наибольшей общей подпоследовательности между строкой $\alpha \in \Sigma^*$ и линеаризациями шаблона π .

Классическая задача поиска наибольшей общей подпоследовательности хорошо изучена для случая двух строк [2]. Предлагаемое обобщение позволяет аналогичным образом ввести меру близости между произвольным шаблоном и строкой, позволяя найти наиболее подходящий шаблон из заданного множества $\Pi = \{\pi_1, \dots, \pi_m\}$.

Предлагаемое решение основано на рассмотрении раскраски частичного порядка и полиномиальном сведении к задаче поиска максимальной антицепи [3]. Особо рассмотрен случай бесповторных шаблонов, которые имеют инъективную функцию разметки λ .

В таблице 1 представлена вычислительная сложность предлагаемых решений. Для огрубления полученных оценок сложности можно воспользоваться неравенствами: $|\preceq| \leq |P|^2$, $\text{LCS}(\alpha, \pi) \leq |P|$.

Задача	Бесповторные шаблоны	Произвольные шаблоны
$\alpha \models \pi$	$\mathcal{O}(\alpha \cdot P)$	$\mathcal{O}(\alpha \cdot \preceq)$
$\text{LCS}(\alpha, \pi)$	$\mathcal{O}(\alpha + \text{LCS}(\alpha, \pi) \cdot P ^2)$	$\mathcal{O}(\text{LCS}(\alpha, \pi) \cdot \alpha ^2 P ^2)$

Таблица 1: Вычислительная сложность предложенных алгоритмов.

Полученные алгоритмы реализованы в виде программной библиотеки и покрыты синтетическими тестами. Проведено экспериментальное сравнение скорости работы алгоритмов по сравнению с переборными алгоритмами, которое подтвердило теоретические оценки вычислительной сложности.

Литература

1. Aggarwal C., Han J. Frequent Pattern Mining, pp.443–461. Springer, 2014.
2. Cormen T., Leiserson C., Rivest R., Stein C. Introduction to Algorithms, 3rd ed., pp.390–397. The MIT Press, 2009.
3. Felsner S., Raghavan V., Spinrad J. Recognition algorithms for orders of small width and graphs of small Dilworth number // Order. 2003. Vol. 20, № 4, pp.351–364.