ОБОЗРЕНИЕ
ПРИКЛАДНОЙ И ПРОМЫШЛЕННОЙ
Том 21 МАТЕМАТИКИ Выпуск 5
2014

Г. Б. Маршалко (Москва, ТВП). **О свойствах одной однонаправленной функции хэширования, основанной на нейронных сетях.**

В работе [1] была предложена схема однонаправленной функции хэширования, основанной на использовании трехслойной нейронной сети и кусочно-линейной функции, с помощью которой определяется преобразование, реализуемое каждым нейроном.

Хэш-функция обрабатывает сообщения блоками по 1024 бита, представленными в виде 32-х 32-битных подблоков $\overline{P} = (P_0, \ldots, P_{31})$, и вырабатывает хэш-код длины 128 бит (4 блока по 32 бита) $\overline{H} = (H_0, H_1, H_2, H_3)$. Перед обработкой 32-битные блоки данных преобразуются в вещественные числа посредством деления на 2^{32}, над которыми и проводятся последующие преобразования.

Базовое кусочно-линейное преобразование определяется выражением (см. рис. 1)

$$f(x) = f(x, Q) = \begin{cases} \dfrac{x}{Q}, \ 0 \leqslant x < Q; \\ \dfrac{x - Q}{0,5 - Q}, \ Q \leqslant x < 0,5; \\ \dfrac{1 - Q - x}{0,5 - Q}, \ 0,5 \leqslant x < 1 - Q; \\ \dfrac{1 - x}{Q}, \ 1 - Q \leqslant x \leqslant 1, \end{cases} \tag{1}$$

где $Q \in (0,\ 0{,}5)$.

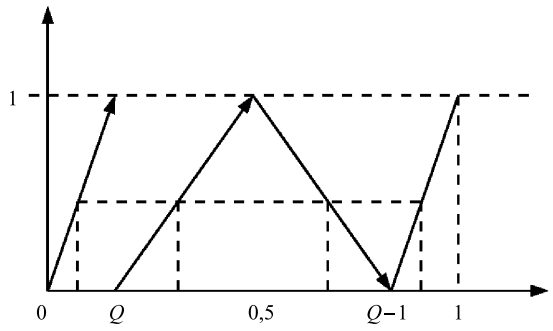

Рис. 1. График функции f

Данное проебразование используется для вычисления значений нейронов в каждом из трех слоев функции хэширования (см. рис. 2).

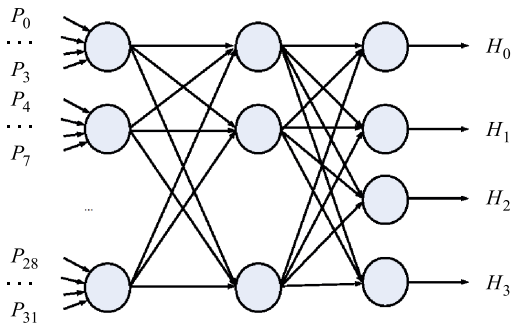

Рис. 2. Структура функции хэширования

Каждый нейрон первого слоя имеет четыре входа и один выход. На вход нейронов первого слоя поступают (32-битные) блоки исходного сообщения (таким образом, в первом слое — 8 нейронов). Каждый выход нейронов первого слоя поступает на вход каждого нейрона второго слоя. Всего во втором слое — 8 нейронов, каждый имеет 8 входов. Последний слой состоит из 4 нейронов, каждый из которых имеет 8 входов.

Значения нейронов вычисляются посредством T-кратного итерирования функции (1). Для первого и третьего слоев $T \geqslant 50$, для второго $T = 1$.

Вычисление значение функции, в конечном итоге, может быть описано следующим выражением:

$$\overline{H} = f_2(\overline{W}_2 f_1(\overline{W}_1 f_0(\overline{W}_0 \overline{P} + \overline{B}_0) + \overline{B}_1)\overline{B}_2). \tag{2}$$

Значения векторов $\overline{B}_i, \overline{W}_i$, а также $\overline{Q}_i, i = 0, 1, 2$ зависят от некоторого параметра, известного только легитимному пользователю.

Рассмотрим подробнее функцию f_0 (здесь f^T — T-кратное итерирование функции f):

$$f_0(\overline{W}_1 \overline{P} + \overline{B}_0) = \begin{pmatrix} f^T(\sum_{i=0}^{3} W_{0,i} P_i + B_{0,0}, Q_{0,0}) \\ f^T(\sum_{i=4}^{7} W_{0,i} P_i + B_{0,1}, Q_{0,1}) \\ \dots \\ f^T(\sum_{i=28}^{31} W_{0,i} P_i + B_{0,7}, Q_{0,7}) \end{pmatrix} \tag{3}$$

В работе [1] указано, что описанное отображение является однонаправленным, в частности для такого отображение невозможно подобрать второе входное сообщение, которое имеет такое же выходное сообщение, как и исходное, быстрее чем за 2^{128} вычислений значения функции. Однако, специфической особенностью рассматриваемого преобразования является то, что исходное сообщение подается на вход нейронов поблочно. Этот факт может быть использован для поиска такого сообщения, в независимости от значения неизвестного параметра схемы.

Действительно, рассмотрим нейрон первого слоя с номером 0. На его вход поступают блоки сообщения $P_0, \dots, P_3$. Выберем P'_0 и P''_0 такие, что $f^T(W_{0,0}P'_0 + \sum_{i=1}^{3} W_{0,i}P_i + B_{0,0}, Q_{0,0}) = f^T(W_{0,0}P''_0 + \sum_{i=1}^{3} W_{0,i}P_i + B_{0,0}, Q_{0,0})$. Для выполнения последнего равенства достаточно потребовать выполнения аналогичного равенства для одной (первой) итерации функции f : $f(W_{0,0}P'_0 + \sum_{i=1}^{3} W_{0,i}P_i + B_{0,0}, Q_{0,0}) = f(W_{0,0}P''_0 + \sum_{i=1}^{3} W_{0,i}P_i + B_{0,0}, Q_{0,0})$. Исходя из задания функции f можно ожидать, что каждому значению функции $y, 0 < y < 1$ соответствует (пренебрегая ошибками, которые могут возникать при преобразовании типов данных) четыре значения аргумента, значению $y = 1$ — два значения ($x = 0,5$ и $x = 1$), значению $y = 0$ — три значения ($x = 0$, $x = Q$ и $x = 1 - Q$). Исходя из этого мы можем предложить следующий способ поиска требуемого сообщения. Выберем значение вектора $\overline{P}$ и зафиксируем значение $P'_0 = P_0$, далее начинаем последовательно (увеличивая каждый раз предыдущее значение на 1) перебирать значения первого блока P''_0 и вычислять

значение функции от вектора $\overline{P}'' = (P_0'', P_2, \ldots, P_{31})$. При этом максимальное расстояние между аргументами, дающими одинаково значение функции не превосходит $\max((1-2Q)\cdot 2^{32}, Q2^{32})$. Отметим, что в данном случае мы не сможем сократить перебор, поскольку нам неизвестны значения параметров $\overline{W}, \overline{B}, \overline{Q}$. В среднем мы найдем требуемый блок через $0,5 \cdot 2^{32}$ шагов, что существенно меньше, чем 2^{128}. Еще раз отметим, что для построения двух входных блоков с одинаковыми выходными значениями функции нам не потребовалось знание значения неизвестого параметра.

СПИСОК ЛИТЕРАТУРЫ

1. *Shiguo Lian*, *Jinsheng Sun*, *Zhiquan Wang.* One-way Hash Function Based on Neural Network. — arxiv.org. 2007.