Об аппроксимации плотного графа разреженным при помощи рандомизированного алгоритма

Селезнёв Михаил Васильевич

Студент

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

E-mail: simplymike8@gmail.com

Научный руководитель — *Ульянов Владимир Васильевич*

Теория графов имеет много приложений — в информатике и программировании, в коммуникационных и транспортных системах, в схемотехнике, в химии. Часто удобнее использовать разреженные графы — например, чтобы минимизировать количество соединений в схеме или построить компьютерную сеть при ограниченном количестве линий связи. Естественным образом возникает вопрос — нельзя ли приблизить граф с большим количеством рёбер разреженным графом?

Пусть мы рассматриваем ориентированный взвешенный граф $G = (V, E, W)$ с $|V| = n$ вершинами и $|E| = m$ рёбрами. Граф можно представить с помощью матрицы Кирхгофа L, которая в свою очередь выражается через матрицу инцидентности рёбер и вершин B и матрицу весов W:

$$L = B^T \cdot W \cdot B = A^T \cdot A$$

где $A = W^{1/2} \cdot B \in \mathbb{R}^{m \times n}$— *взвешенная матрица инцидентности.* В плотных графах $m \sim n^2$, и при большом количестве вершин $m \gg n$. Хотелось бы уменьшить это соотношение. Тогда задача сводится к тому, чтобы найти некоторое приближение $\tilde{A} \in \mathbb{R}^{\tilde{m} \times n}$ этой матрицы, где $\tilde{m} \ll m$. При этом мы хотим, чтобы матрица $\tilde{A}$ задавала разреженный подграф исходного графа, т. е. была корректной взвешенной матрицей инцидентности, причём её строки были бы строками матрицы A (возможно, домноженными на положительную константу, что соответствует изменению веса ребра).

В некоторых более ранних работах на схожую тему (например, [1]) было показано, что существует такое распределение строк, что, выбирая из него $O(n \log n)$ строк, мы с высокой вероятностью получим хороше приближение в евклидовой норме, т. е.:

$$(1 - \varepsilon)\|Ax\|_2 \leqslant \|Bx\|_2 \leqslant (1 + \varepsilon)\|Ax\|_2, \quad \forall x \in \mathbb{R}^n.$$

Однако, проблема заключается в том, что для нахождения этого распределения нужно иметь либо матрицу $A^T A$, либо её приближение, что возвращает нас к проблеме выбора строк.

Предлагается использовать подход, предложенный авторами работы [2].

$$
\begin{tikzcd}
\boxed{\mathbf{A} = \mathbf{A}(0)} \arrow[rr] & & \boxed{\mathbf{A}(1)} \arrow[r] & \cdots \arrow[rr] & & \boxed{\mathbf{A}(L-1)} \arrow[rr] & & \boxed{\mathbf{A}(L)} \\
\boxed{\mathbf{B} = \mathbf{B}(0)} \arrow[u, phantom, "\wr"] & & \boxed{\mathbf{B}(1)} \arrow[u, phantom, "\wr"] \arrow[dl] & \cdots & & \boxed{\mathbf{B}(L-1)} \arrow[u, phantom, "\wr"] \arrow[dl] & & \boxed{\mathbf{B}(L)} \arrow[u, phantom, "\|"] \arrow[dl] \\
& \boxed{p(1)} \arrow[ul] & & \cdots \arrow[ul] & \cdots & & \boxed{p(L)} \arrow[ul] &
\end{tikzcd}
$$

Рис. 1

Он проиллюстрирован на рисунке 1. Сначала, используя метод случайных проекций или просто случайным образом выбирая строки, мы получаем последовательность $A(l)$ грубых приближений всё меньшей размерности. Это соответствует движению вправо на диаграмме. Дойдя до матрицы достаточно малого размера, можно приступить к движению в обратную сторону. Используя небольшую матрицу $B(l+1)$, являющуюся неплохим приближением $A(l)$, мы оцениваем вероятности, с которыми надо выбирать строки $A(l)$, чтобы получить её приближение $B(l)$.

Если на каждом шаге при движении влево на диаграмме оставлять в новой матрице половину строк, то в среднем за $O(\log n)$ шагов мы придём к матрице $A(L)$ с $O(n \log n)$ строками. На каждом шаге для оценки вероятностей строк нужно будет считать произведение $B^T(l+1) \cdot B(l+1)$, что может быть сделано за $O(n^3 \log^2 n)$ (так как в B ровно n столбцов и порядка $n \log n$ строк). Оценка вероятностей строк при наличии приближения с $n \log n$ строками может быть осуществлена за $O(mn \log n)$. Выбор $\tilde{m}$ строк в соответствии с полученным распределением реализуется со сложностью $O(m + \tilde{m} \log m)$.

Таким образом, учитывая, что в плотном графе $m \sim n^2$ и целясь в результат $\tilde{m} = O(n \log n)$, можем построить алгоритм, имеющий рандомизированную сложность

$$O\big((n^3 \log^2 n + n^3 \log n + n^2 + 2n \log^2 n) \cdot \log n\big) = \boxed{O(n^3 \log^3 n).}$$

Для сравнения, наивный алгоритм, честно вычисляющий матрицу $A^T A$, тратил бы на это время $O(nm^2) = O(n^5)$.

Литература

1. M. Rudelson, R. Vershynin, Sampling from large matrices: An approach through geometric functional analysis. // J. ACM, 54(4):21, 2007.
2. Mu li, Gary L. Miller, Richard Peng. Iterative Row Sampling. // April 13, 2013