Оглавление

*20.3. Максимизация влияния

`GreedyCoverage` (раздел 20.2) изначально задумывался для классических применений, таких как выбор местоположений для новых заводов. В XIX веке обобщения этого алгоритма нашли новое применение в computer science. Этот раздел описывает репрезентативный пример в анализе социальных сетей.[1]

20.3.1. Каскады в социальных сетях

Для наших целей *социальная сеть* — это ориентированный граф $G = (V, E)$, в котором вершины соответствуют пользователям, а ориентированное ребро (v, w) означает, что v «влияет на» w. Например, w следит за новостями v.

ПРОСТАЯ КАСКАДНАЯ МОДЕЛЬ

Изначально каждая затравочная вершина является «активной», а все остальные вершины «неактивными». Все ребра изначально находятся «не в подброшенном» состоянии.

Пока есть активная вершина v и не подброшенное исходящее ребро (v, w), надо:

- подбросить несмещенную монету: орел выпадает с вероятностью p;
- нсли выпадает орел, обновить статус ребра (v, w) значением «активное». Если вершина w не активна, обновить статус ребра значением «активное»;
- если выпадает решка, обновить статус ребра (v, w) значением «неактивное».

Каскадная модель показывает, как информация (например, новостная статья или мем) перемещается по социальной сети. Вот простой пример сети,

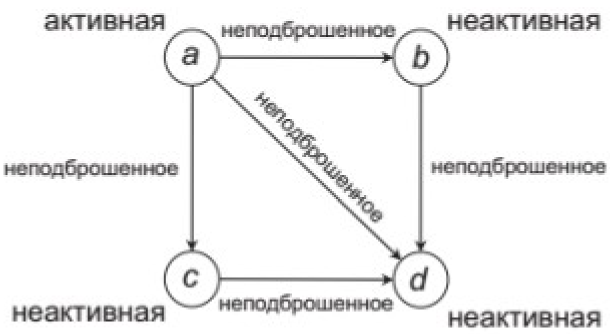
[1] Разделы, помеченные звездочкой, подобные этому, являются более трудными, и при первом прочтении их можно пропустить.

параметризованной ориентированным графом $G = (V, E)$, активационной вероятностью $p \in [0, 1]$ и подмножеством $S \subseteq V$ *затравочных* (seed) вершин.[1]

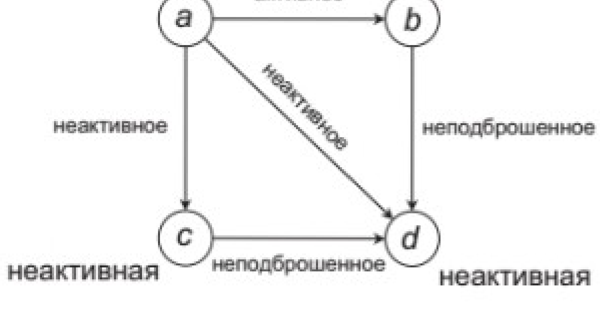
После того как вершина активирована (скажем, вследствие чтения статьи или просмотра фильма), она больше не станет неактивной. Вершина может иметь несколько активационных возможностей — одну для каждого ее активизированного влиятельного объекта. Например, первые две рекомендации от друзей относительно нового фильма не регистрируются, но третья побуждает вас его посмотреть.

20.3.2. Пример

В графе

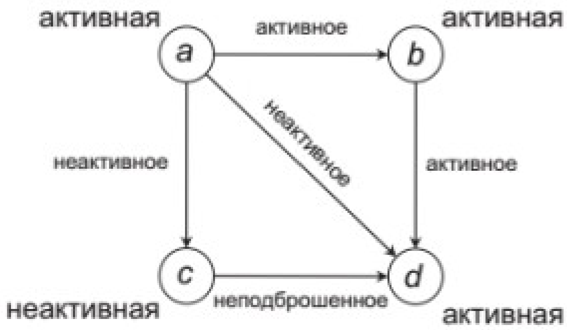


вершина a является затравочной и изначально активной. Остальные вершины изначально неактивны. Каждое из исходящих ребер (a, b), (a, c) и (a, d) имеет вероятность p активирования другой конечной точки ребра. Предположим, что монета, связанная с ребром (a, b), выпадает орлом, а две другие — решкой. Новая картина такова:

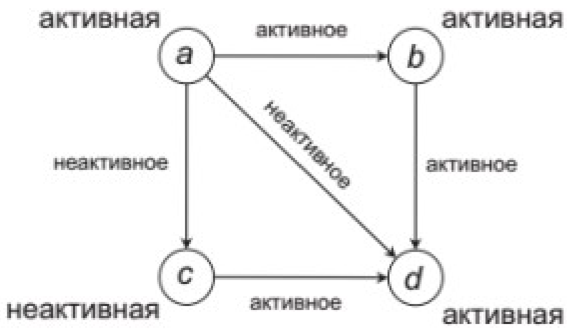


[1] Для того чтобы освежить информацию о базовой дискретной вероятности, см. Приложение Б *Первой части* либо ресурсы на веб-сайте www.algorithmsilluminated.org.

В этот момент нет никакой надежды активировать вершину c. Остается вероятность p, что вершина d активируется через неподброшенное ребро (b, d). Если это событие произойдет, то финальным состоянием будет:



Для замыкания и удобства можно факультативно добавить шаг постобработки, который подбрасывает монеты любых оставшихся неподброшенных ребер и соответствующим образом обновляет их статусы (оставляя статусы всех вершин неизменными). В нашем рабочем примере окончательный результат может быть таким:



В общем случае, с шагом постобработки или без него, вершины, которые в итоге активируются в конце процесса, достигаются из затравочной вершины по ориентированному пути, состоящему из активированных ребер.

20.3.3. Задача о максимизации влияния

Цель задачи о *максимизации влияния* состоит в выборке ограниченного числа затравочных вершин в социальной сети для максимизации разброса информации (при числе вершин, которые в конечном итоге активируются в соответствии с каскадной моделью).[1] Это число является случайной величиной, зависящей от исходов подбрасываний монеты в каскадной модели, и мы сосредоточимся на его математическом ожидании.[2] Формально обозначим через $X(S)$ (случайное) множество вершин, которые в конечном итоге активируются, когда вершины S выбираются в качестве затравочных, и определим *влияние* S как

$$f_{\text{inf}}(S) = \mathbf{E}\,[|X(S)|], \tag{20.10}$$

где математическое ожидание строится над случайными подбрасываниями монеты в каскадной модели. Влияние множества зависит как от графа, так и от активационной вероятности, причем большее число ребер или более высокая вероятность приводят к большему влиянию.

ЗАДАЧА: МАКСИМИЗАЦИЯ ВЛИЯНИЯ

Вход: ориентированный граф $G = (V, E)$, вероятность p и положительное целое число k.

Выход: выборка $S \subseteq V$ из k вершин с максимально возможным влиянием $f_{\text{inf}}(S)$ в каскадной модели с активационной вероятностью p.

Например, если вы раздаете k рекламных копий продукта и хотите выбрать получателей, чтобы максимизировать его принятие, то вы сталкиваетесь с задачей о максимизации влияния.

Задача 20.9 просит вас показать, что задача о максимальном охвате может рассматриваться как частный случай задачи о максимизации влияния. Поскольку частный случай является NP-трудным (задача 22.8), то и более общая

---

[1] Подробнее о задаче о максимизации влияния и ее многочисленных вариациях читайте в статье Дэвида Кемпе, Джона Клейнберга и Эвы Тардос «Максимизирование распространения влияния через социальную сеть» («*Maximizing the Spread of Influence Through a Social Network», David Kempe, Jon Kleinberg, Éva Tardos, Theory of Computing*, 2015).

[2] Математическое ожидание $\mathbf{E}\,[Y]$ случайной величины Y — это ее среднее значение, взвешенное соответствующими вероятностями. Например, если Y может принимать значения $\{0, 1, 2, \ldots, n\}$, то $\mathbf{E}[Y] = \sum_{i=0}^{n} i \times \mathbf{Pr}[Y = i]$.

задача тоже. Может ли существовать быстрый и приближенно правильный эвристический алгоритм для задачи о максимизации влияния?

20.3.4. Жадный алгоритм

Задача о максимизации влияния напоминает задачу о максимальном охвате, где вершины играют роль подмножеств, а влияние (20.10) играет роль охвата (20.5). `GreedyCoverage` для последней задачи подходит для первой, если подставить новое определение (20.10) целевой функции.

GREEDYINFLUENCE

Вход: ориентированный граф $G = (V, E)$, вероятность $p \in [0, 1]$ и положительное целое число k.

Выход: множество $S \subseteq V$ из k вершин.

$S := \emptyset$ `// выбранные вершины`

for $j = 1$ to k **do** `// выбирать вершины одну за другой`

  `// жадно увеличивать влияние`

  `// (произвольно разрывать связи)`

  $v^* := \text{argmax}_{v \in V}[f_{\text{inf}}(S \cup \{v\}) - f_{\text{inf}}(S)]$

  $S := S \cup \{v^*\}$

return S

ТЕСТОВОЕ ЗАДАНИЕ 20.6

Каково время выполнения простой реализации `GreedyInfluence` на графах с n вершинами и m ребрами? (Выбрать самый точный ответ.)

а) $O(knm)$

б) $O(knm^2)$

в) $O(knm^{2m})$

г) неясно

(Ответ и анализ решения см. в разделе 20.3.8.)
