Скрытые марковские модели

Семинар был посвящен обсуждению нового для нас аппарата скрытых марковских моделей.

Скрытая марковская модель (СММ) — статистическая модель, имитирующая работу процесса, похожего на марковский процесс с неизвестными параметрами, и задачей ставится разгадывание неизвестных параметров на основе наблюдаемых. Полученные параметры могут быть использованы в дальнейшем анализе, например, для распознавания образов.

В скрытой марковской модели мы можем следить лишь за переменными, на которые оказывает влияние данное *скрытое* состояние цепи Маркова. Каждое состояние имеет вероятностное распределение среди конечного множества возможных наблюдаемых состояний.

Определим параметры, описывающие поведение СММ.

- T - длина последовательности наблюдений,
- N - количество скрытых состояний цепи,
- M - количество наблюдаемых состояний,
- $Q = \{q_1, q_2, \ldots, q_T\}$ - последовательность скрытых состояний цепи (q_t - скрытое состояние цепи в момент времени n),
- $V = \{v_1, v_2, \ldots, v_M\}$ - множество наблюдаемых состояний,
- $A = \{a_{ij}\}, a_{ij} = \mathbf{P}(q_{t+1} = S_j \mid q_t = S_i)$ - матрица переходных вероятностей скрытой ЦМ,
- $B = \{b_j(k)\}, b_j(k) = \mathbf{P}(v_k \text{ в момент } t \mid q_t = S_i)$ - вероятность наблюдения (эмиссии) из состояния j,
- $\pi = \{\pi_i\}, \pi_i = \mathbf{P}(q_1 = S_i)$ - начальное распределение цепи.

Последовательность наблюдений обозначается $O = \{O_1, \ldots, O_T\}$. Введем так же обозначение $\lambda = (A, B, \pi)$.

Для СММ можно сформулировать 3 основные задачи.

Задача 1 По известным наблюдениям $O = \{O_1, \ldots, O_T\}$ и параметрам $\lambda = (A, B, \pi)$ найти вероятность $\mathbf{P}(O \mid \lambda)$.

Задача 2 По известным наблюдениям $O = \{O_1, \ldots, O_T\}$ и параметрам $\lambda = (A, B, \pi)$ найти наиболее оптимальную (в некотором смысле) последовательность скрытых состояний $Q = \{q_1, q_2, \ldots, q_N\}$, стоящую за данными наблюдениями.

Задача 3 По известным наблюдениям $O = \{O_1, \ldots, O_T\}$ подобрать параметры $\lambda = (A, B, \pi)$, максимизирующие $\mathbf{P}\left(O \mid \lambda\right)$.

Проиллюстрируем, как эти задачи могут применяться в распознавании речи.

Пример. Пусть W - словарь слов, каждое из которых мы должны научиться распознавать из звукового сигнала. Для этого для каждого слова $w \in W$ мы создаем СММ с N скрытыми состояниями. Чтобы обучить каждую из них, в качестве наблюдаемой последовательности нужно использовать показания, принимающие одно из M значений. Чтобы преобразовать исходный звуковой сигнал в такую дискретную последовательность, проделывают процедуру *квантования* сигнала. В результате процедуры получается выборка длины T, каждый элемент которой принимает одно из M значений.

Имея теперь выборку наблюдений, можно решить задачу 3 и для каждой СММ подобрать набор параметров λ_w.

Далее, решая задачу 2 для построенных СММ, можно придавать смысл скрытым состояниям и более тонко настраивать модель: например, менять количество скрытых состояний.

Теперь, имея обученные СММ для каждого слова из словаря, для нового поступившего сигнала можно находить вероятности $\mathbf{P}(O \mid \lambda_w)$ для всех слов w из словаря (это задача 1), и считать моделью для сигнала то слово, для которого эта вероятность максимальна.

На семинаре мы рассмотрели алгоритмы решения первой задачи, так называемый алгоритм прямого-обратного хода.

Решение задачи 1. Алгоритм прямого-обратного хода

Введем переменную прямого хода

$$\alpha_t(i) = P\left(O_1 O_2 \cdots O_t, q_t = S_i \mid \lambda\right).$$

Ее значения для всех $t = 1, \ldots, T$ и $i = 1, \ldots, N$ можно найти итеративно:

1. Инициализация: $\alpha_1(i) = \pi_i b_i\left(O_1\right), \quad 1 \le i \le N$;
2. Индукция: $\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij}\right] b_j\left(O_{t+1}\right), \quad 1 \le t \le T-1$ и $1 \le j \le N$.

Искомую вероятность теперь можно найти как

$$\mathbf{P}(O \mid \lambda) = \sum_{i=1}^N \alpha_T(i).$$

Трудоемкость алгоритма составляет $O(N^2T)$.

Используя аналогичный подход (динамического программирования) можно решать и задачи 2 и 3. Алгоритм решения второй задачи называется алгоритмом Витерби. Третью задачу решают алгоритмом Баума-Уэлша.