Машинное обучение, ФКН ВШЭ
Семинар №22

Мы уже не раз сталкивались с идеей поиска *представлений* для объектов — например, слов (word2vec) или изображений (последние слои свёрточных сетей). В этой лекции мы обсудим задачу поиска представлений для текстов.

Текст может кодироваться с помощью вектора, каждый элемент которого соответствует одному слову из словаря, а его значение вычисляется как число вхождений этого слова в текст или, например, как TF-IDF. Впрочем, такой подход никак не учитывает наличие синонимов (было бы полезно объединять слова по смыслу), многозначных слов (которые, наоборот, не следует объединять в одну координату) и других смысловых особенностей. Чтобы работать на уровне смыслов, предположим, что существует T *тем* (topics), и каждый документ x_d характеризуется вектором $\theta_d \in \mathbb{R}^T$. Элементы этого вектора должны описывать наличие той или иной темы в данном документе. Далее можно рассматривать эти векторы как новые признаковые описания документов, характеризующие его уже на уровне смыслов, а не слов. Темой называется вектор $\varphi_t \in \mathbb{R}^W$, где W — размер словаря; этот вектор должен характеризовать принадлежность каждого слова к данной теме. Ниже мы рассмотрим несколько подходов к построению таких *тематических моделей*

1 Latent semantic analysis (LSA)

Построим матрицу $X \in \mathbb{R}^{D\times W}$, где D — число документов, W — размер словаря. С помощью сингулярного разложения найдём лучшую аппроксимацию (в смысле квадратичного отклонения) ранга T:

$$X = \Theta\Phi, \quad \Theta \in \mathbb{R}^{D\times T},\ \Phi \in \mathbb{R}^{T\times W}.$$

Строки матрицы Θ можно интерпретировать как распределения тем в документах, столбцы матрицы Φ — как распределения слов в темах. Заметим, что эти векторы не являются распределениями в прямом смысле, поскольку их элементы могут быть отрицательными. Такие представления могут быть полезны для понижения размерности или для учёта смысловых близостей слов, но не поддаются интерпретации.

2 Probabilistic latent semantic analysis (PLSA)

Будем считать, что каждый документ x_d описывается распределением $p(t \mid d) = = \theta_{td}$, а каждая тема — распределением $p(w \mid t) = \varphi_{wt}$. Тогда совместное распределе-

ние на словах и документах можно записать как

$$p(w,d) = p(d)p(w \mid d) = p(d)\sum_{t=1}^{T} p(w \mid t)p(t \mid d).$$

Здесь мы, по сути, ввели скрытую переменную t, которая показывает, из какой темы было сгенерировано слово w документа x_d. Согласно данной модели, документ x_d генерируется по следующей схеме:

1. Выбираем тему $t \sim p(t \mid d)$;
2. Выбираем слово из данной темы $w \sim p(w \mid t)$;
3. Повторяем шаги 1 и 2, если текст не достиг требуемой длины.

Чтобы записать правдоподобие, следует смотреть на набор документов как на выборку пар «документ-слово». Неполное правдоподобие данной модели имеет вид

$$\sum_{d=1}^{D}\sum_{j=1}^{|x_d|}\log\sum_{t=1}^{T}\varphi_{w_{dj}t}\theta_{td},$$

где w_{dj} — j-е по порядку слово из документа x_d. Если для каждой пары «документ-слово» (d, w_{dj}) известно, из какой темы t_{dj} оно сгенерировано, то можно записать полное правдоподобие:

$$\sum_{d=1}^{D}\sum_{j=1}^{|x_d|}\sum_{t=1}^{T}[t_{dj}=t]\log\varphi_{w_{dj}t}\theta_{td}.$$

Мы уже знаем, что для обучения таких моделей можно воспользоваться EM-алгоритмом. На E-шаге оценим апостериорные распределения на скрытых переменных по формуле Байеса:

$$p(t_{dj} \mid d, w_{dj}) = \frac{p(w_{dj} \mid t_{dj})p(t_{dj} \mid d)}{p(w_{dj} \mid d)} = \frac{\varphi_{w_{dj}t_{dj}}\theta_{t_{dj}d}}{p(w_{dj} \mid d)}$$

На M-шаге найдём максимум матожидания полного правдоподобия по скрытым переменным:

$$\varphi_{wt} = \frac{\sum_{d=1}^{D} n_{dw}p(t \mid d, w)}{\sum_{w=1}^{W}\sum_{d=1}^{D} n_{dw}p(t \mid d, w)};$$
$$\theta_{td} = \frac{\sum_{w=1}^{W} n_{dw}p(t \mid d, w)}{\sum_{t=1}^{T}\sum_{w=1}^{W} n_{dw}p(t \mid d, w)}.$$

Здесь n_{dw} — число вхождений слова w в документ x_d.

Полученная в итоге работы EM-алгоритма модель будет интепретируемой — можно изучать, насколько сильно та или иная тема представлена в документе, или насколько то или иное слово характерно для темы.

3 Latent Dirichlet Allocation (LDA)

Модель PLSA не является полной — распределения φ_t и θ_d нужно заранее задать. Значит, с помощью данной модели не получится описать процесс порождения набора документов «с нуля». Более того, в PLSA отсутствует регуляризация, из-за чего модель может слишком сильно подогнаться под данные на небольших выборках.

Чтобы устранить два указанных недостатка, введём априорные распределения на векторах φ_t и θ_d. Для этого хорошо подходит симметричное распределение Дирихле, которое задано на множестве всех дискретных распределений с фиксированным числом исходов:

$$\varphi_t \sim \mathrm{Dir}(\alpha);$$
$$\theta_d \sim \mathrm{Dir}(\beta);$$
$$\mathrm{Dir}(x_1, \ldots, x_n; \alpha) = \frac{\Gamma(\alpha n)}{\Gamma(\alpha)^n} \prod_{i=1}^{n} x_i^{\alpha-1}.$$

После введения априорного распределения модель становится достаточно сложной для вывода. Как правило, для её обучения применяют техники вариационного вывода или семплирование Гиббса.