Прикладные задачи анализа данных

Семинар 5

Модели скрытых тем

Национальныи Исследовательский Университет
Высшая Школа Экономики

15 февраля 2018

Задача представления текста

- Можно представить текст с помощью вектора, каждый элемент которого соответствует одному слову из словаря и вычисляется как число вхождения этого слова в текст или TF-IDF ("мешок слов").
- Такой подход не учитывает наличие синонимов или многозначных слов, т.е. не позволяет учитывать смысл текста.
- Эту проблему решает способ представления документов коллекции с помощью векторов, где каждый элемент вектора характеризует принадлежность документа к какой-то теме.

Типичные приложения:

- анализ коллекций научных статей
- анализ новостных потоков (рубрикация)
- аннотация генома и другие задачи биоинформатики
- коллаборативная фильтрация

Определения

- **Тематическое моделирование** — способ построения модели коллекции текстовых документов, которая определяет, к каким темам относится каждый из документов.
- **Тема** — набор терминов (униграм или n—грам) часто встречающихся вместе в документах. (Интуитивно: набор слов, глядя на которые можно сказать, какую предметную область они описывают.
- Тематическая модель исследует скрытую тематическую структуру коллекции текстов:
  - **Тема** ϕ_t — это вероятностное распределение $p(w|t)$ над терминами w
  - **Документ** x_d — это вероятностное распределение $p(t|d)$ над темами ϕ_t

Посмотрим более пристально:

- **Тема** ϕ_t — это вероятностное распределение $p(w|t)$ над терминами w.
  - это вектор размерности, равной размеру всего словаря W
  - этот вектор характеризует принадлежность каждого слова к данной теме
- **Документ** x_d — это вероятностное распределение $p(t|d)$ над темами ϕ_t.
  - это вектор размерности, равной количеству тем T
  - этот вектор описывает наличие темы в данном тексте

Подходы к построению тематических моделей

- PLSA – простейшая модель без регуляризации
- LDA - байесовская модель со сглаживанием

pLSA = Latent semantic analysis

- Рассмотрим матрицу $X \in R^{D \times W}$, где D — число документов, W — размер словаря.
- Найдем аппроксимацию с помощью сингулярного разложения ранга T:

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

.

- 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 учета смысловых близостей слов, но не поддаются интерпретации.

PLSA = Probabilistic latent semantic analysis — 1

- Итак, каждый документ x_d описывается распределением $p(t|d) = \theta_{td}$, а каждая тема — распределеним $p(w|t) = \phi_{wt}$
- Тогда совместное распределение на словах и документах можно записать так:

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

.

- Т.е. мы ввели **скрытую** переменную t, которая показывает, из какой темы было сгенерировано слово w документа x_d

PLSA = Probabilistic latent semantic analysis — 2

Согласно данной модели документ x_d генерируется по следующей схеме:

- Выбираем тему $t \sim p(t|d)$
- Выбираем слово из данной темы $w \sim p(w|t)$
- Повторяем первые два шага, если текст не достиг требуемой длины.

PLSA = Probabilistic latent semantic analysis — 3

- Чтобы записать правдоподобие, следует смотреть на набор документов как на пару "документ-слово".
- Обозначим за w_{dj} — j-тое по порядку слово из документа x_d
- Если для каждой пары "документ-слово"(d, w_{dj}) известно, из какой темы t_{dj} оно сгенерировано, можно записать полное правдоподобие:

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

PLSA = Probabilistic latent semantic analysis — 4

Для обучения таких моделей пользуются EM-алгоритмом:

- E-шаг: оцениваем апостериорные распределения на скрытых переменных $p(t_{dj}|d, w_{dj})$
- M-шаг: находим максимум матожидания полного правдоподобия по скрытым переменным ϕ_{wt}, θ_{td}.

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

Недостатки PLSA:

- Модель PLSA не является полной: распределения ϕ_t, θ_d нужно заранее задать. Т.е. не получится описать процесс порождения набора документов "с нуля".
- В PLSA отсутствует регуляризация, т.е. модель может слишком подогнаться под данные на небольших выборках.

- Введем априорные распределения на векторах ϕ_t, θ_d.
- Для этого подходит распределене Дирихле, которое задано на множестве всех дискретных распределений с фиксированным число исходов:

$$\phi_t \sim Dir(\alpha)$$

$$\theta_d \sim Dir(\beta),$$

где

$$Dir(x_1, \ldots, x_n; \alpha) = \frac{\Gamma(\alpha n)}{\Gamma(\alpha)^n} \prod_{i=1}^{n} x_i^{\alpha-1}$$

- Распределение Дирихле удобно еще и тем, что позволяет управлять разреженностью Φ, Θ.

EM-алгоритм "на пальцах— 1

N раз выбираем синюю или красную монету и делаем 3 броска.
Наблюдаем:

- $X = (x_{11}, x_{12}, x_{13}) \dots (x_{N1}, x_{N2}, x_{N3})$ — исходы бросков
- $T = t_1, \dots t_N$ — цвета выбранных меток

Хотим узнать:

- $\lambda = p(blue) =$
- $\theta_1 = p(head|blue) =$
- $\theta_2 = p(head|red) =$

EM-алгоритм "на пальцах— 2

N раз выбираем синюю или красную монету и делаем 3 броска.

Наблюдаем:

- $X = (x_{11}, x_{12}, x_{13}) \dots (x_{N1}, x_{N2}, x_{N3})$ — исходы бросков
- $T = t_1, \dots t_N$ — цвета выбранных меток

Хотим узнать:

- $\lambda = p(blue) = \frac{bluethrow}{N}$
- $\theta_1 = p(head|blue) = \frac{head,blue}{blue}$
- $\theta_2 = p(head|red) = \frac{head,red}{red}$

N раз выбираем синюю или красную монету и делаем 3 броска.
Наблюдаем:

- $X = (x_{11}, x_{12}, x_{13}) \dots (x_{N1}, x_{N2}, x_{N3})$ — исходы бросков
- $T = t_1, \dots t_N$ — цвета выбранных меток

Метод максимума правдоподобия:

$$\ln p(X, T|\Theta) = \sum_{i=1}^{N} \ln p(x_i, t_i|\Theta) = \sum_{i=1}^{N} p(x_i|t_i, \Theta)p(t_i|\Theta) \to \max_{\Theta=\lambda,\theta_1,\theta_2}$$

N раз выбираем синюю или красную монету и делаем 3 броска.
Наблюдаем:

- $X = (x_{11}, x_{12}, x_{13}) \dots (x_{N1}, x_{N2}, x_{N3})$ — исходы бросков

От нас скрыты:

- $T = t_1, \dots t_N$ — цвета выбранных меток

Хотим узнать:

- $\lambda = p(blue) =$
- $\theta_1 = p(head|blue) =$
- $\theta_2 = p(head|red) =$

N раз выбираем синюю или красную монету и делаем 3 броска.
Наблюдаем:

- $X = (x_{11}, x_{12}, x_{13}) \dots (x_{N1}, x_{N2}, x_{N3})$ — исходы бросков

От нас скрыты:

- $T = t_1, \dots t_N$ — цвета выбранных меток

Метод максимума правдоподобия:

$$\ln p(X|\Theta) = \sum_{i=1}^{N} \ln p(x_i|\Theta) = \sum_{i=1}^{N} \ln \sum_{t_i} p(x_i, t_i|\Theta) \to \max_{\Theta=\lambda,\theta_1,\theta_2}$$

Известные реализации:

- Gensim (Online Variation LDA, Python, parallel)
- BigARTM (Online ARTM, C++/Python, parallel)
- Vowpal Wabbit LDA (Online Variation LDA, C++)
- Scikit-learn LDA (Online Variation LDA, Python)

Задание семинара

Скачиваем тут

При подготовке семинара использовались

- материалы лекции Евгения Соколова по курсу МО на ФКН ПМИ
- материалы семинара Анны Потапенко по курсу МО в ШАД
- материалы семинара Мурата Апишева по курсу АНД на ФКН ПМИ