Векторные представления слов

Математические методы анализа текстов
осень 2021

Попов Артём Сергеевич

28 сентября 2021

Задача разметки последовательности (sequence tagging)

Дано D — множество размеченных последовательностей (x, y):

- $x = \{x_1, \dots, x_n\}$ — входная последовательность (слова)
- $y = \{y_1, \dots, y_n\}$ — выходная последовательность (метки)
- $y_i \in Y$ — метка для $x_i \in X$

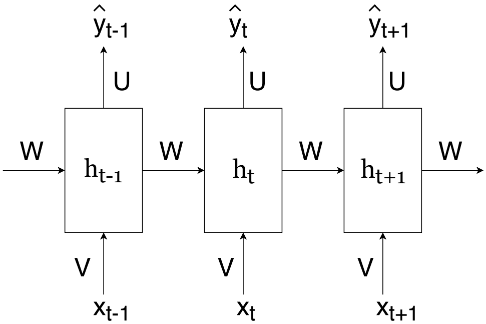
Необходимо по входной последовательности предсказать элементы выходной последовательности.

Длины x и y в одной паре совпадают, но могут различаться в разных парах. Две последовательности можно привести к одной длине дополнив меньшую специальным <PAD> токеном.

Подходы для разметки последовательностей

- Rule-based подход
- Классификатор на каждой позиции, использующий признаки контекста позиции
- Графические модели (HMM / MEMM / CRF)
- Нейронные сети (рекуррентные, трансфомеры, свёрточные)
- Комбинация нейронных сетей и графических моделей

Напоминание. Рекуррентная нейронная сеть.



h_t — скрытое состояние сети в момент времени t

$$h_t = f(Vx_t + Wh_{t-1} + b)$$

$$\hat{y}_t = \mathrm{softmax}(Uh_t + \hat{b})$$

Обучение сети (backpropagation through time + SGD):

$$\sum_{t=1}^{n} \mathcal{L}_t(y_t, \hat{y}_t) \to \min_{V,U,W,b,\hat{b}}$$

[1] Rumelhart et al; Learning internal representations by error propagation; 1985

Применение нейросетевой модели разметки

Этап 1. Вычисление вероятностей меток:

$$\hat{y}_t = RNN(x)_t$$

Этап 2. Преобразование вероятностей в значения:

$$\tilde{y}_t = \arg\max \hat{y}_t$$

Какие могут быть проблемы с этим подходом?

Применение нейросетевой модели разметки

Этап 1. Вычисление вероятностей меток:

$$\hat{y}_t = RNN(x)_t$$

Этап 2. Преобразование вероятностей в значения:

$$\tilde{y}_t = \arg\max \hat{y}_t$$

Какие могут быть проблемы с этим подходом?

Несогласованность выдачи предсказаний для разных меток.

Пример. Возможные, ошибки разметки.

Некоторые ошибки обоснованы несовершенностью модели:

Давай	встретимся	на	библиотеке	имени	Ленина
O	O	O	B-LOC	**B-LOC**	I-LOC
O	O	O	B-LOC	I-LOC	**B-PER**

А некоторые ошибки прямо противоречат постановке задачи:

Давай	встретимся	на	библиотеке	имени	Ленина
O	O	O	B-LOC	I-LOC	**I-PER**
O	O	O	**I-LOC**	I-LOC	I-LOC

Модель Линейного Conditional Random Field (CRF)

Multinomial regression	Linear-CRF
$x \in X, y \in Y$	$x = (x_1, \ldots, x_n),\ x_i \in X$
	$y = (y_1, \ldots, y_n),\ y_i \in Y$

Модель Линейного Conditional Random Field (CRF)

Multinomial regression	Linear-CRF
$x \in X, y \in Y$	$x = (x_1, \dots, x_n),\ x_i \in X$ $y = (y_1, \dots, y_n),\ y_i \in Y$
$a(x, y) = \sum_{j=1}^{d} \theta_j^y f_j(x)$	$a(x, y) = \sum_{j=1}^{d} \theta_j F_j(x, y),$
$\theta^y \in \mathbb{R}^d, \quad f_j(x) \in \mathbb{R}$	$F_j(x, y) = \sum_{i=1}^{n} f_j(y_i, y_{i-1}, x, i)$ $\theta \in \mathbb{R}^d, \quad F_j(x, y), f_j(\dots) \in \mathbb{R}$

Признаки в linear-CRF

$f_j(y_i, y_{i-1}, x, i)$ — информация о последовательности x, полезная для предсказания метки y_i в позиции i, когда в предыдущей позиции $(i-1)$ находится метка y_{i-1}.

- $f_j(\bullet, i)$ может зависеть от всего x, не только от x_i
- $f_j(\bullet, i)$ не может зависеть от других меток, кроме y_{i-1} и y_i
- последовательности (x, y) могут иметь любые длины n, но размерность $F(x, y)$ фиксирована и равна d

Классические примеры признаков для задачи POS

- $y_i = \text{ADVERB}$ и слово x_i оканчивается на «-ly»
  Ожидаем $\theta_j > 0$, такие слова часто оказываются наречиями.
- $y_i = \text{PRONOUN}$ и $i = 1$ и предложение оканчивается знаком «?»
  Ожидаем $\theta_j > 0$, первое слово в вопросительных предложениях часто оказывается местоимением
- $y_i = \text{NOUN}$ и $y_{i-1} = \text{ADJECTIVE}$
  Ожидаем $\theta_j > 0$, существительные часто следуют за прилагательным
- $y_i = \text{PREPOSITION}$ и $y_{i-1} = \text{PREPOSITION}$
  Ожидаем $\theta_j < 0$, два предлога редко встречаются подряд

Модель Линейного Conditional Random Field (CRF)

Multinomial regression	Linear-CRF
Обучение (ММП + SGD):	
$p(y\|x) = \operatorname{softmax}_{y \in Y} a(x, y)$	$p(y\|x) = \operatorname{softmax}_{y \in Y^n} a(x, y)$
$\sum_{(x,y) \in D} \log p(y\|x) \to \min_{\theta}$	$\sum_{(x,y) \in D} \log p(y\|x) \to \min_{\theta}$

Модель Линейного Conditional Random Field (CRF)

Multinomial regression	Linear-CRF

Обучение (ММП + SGD):

$p(y|x) = \operatorname{softmax}_{y \in Y} a(x, y)$ $\qquad$ $p(y|x) = \operatorname{softmax}_{y \in Y^n} a(x, y)$

$\sum_{(x,y) \in D} \log p(y|x) \to \min_{\theta}$ $\qquad$ $\sum_{(x,y) \in D} \log p(y|x) \to \min_{\theta}$

Применение:

$\hat{y} = \arg\max_{y \in Y} p(y|x)$ $\qquad$ $\hat{y} = \arg\max_{y \in Y^n} p(y|x)$

Какие вычислительные проблемы вы заметили?

Вычисление оптимальной разметки. Функция Φ.

Для получения разметки нормировочная константа не нужна:

$$\hat{y} = \arg\max_{y \in Y^n} p(y|x) = \arg\max_{y \in Y^n} \operatorname{softmax}_{y \in Y^n} a(x, y) = \arg\max_{y \in Y^n} a(x, y)$$

$$a(x, y) = \sum_{j=1}^{d} \theta_j F_j(x, y) = \sum_{j=1}^{d} \theta_j \sum_{i=1}^{n} f_j(y_i, y_{i-1}, x, i) =$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{d} \theta_j f_j(y_i, y_{i-1}, x, i) \equiv \sum_{i=1}^{n} \Phi_i(y_i, y_{i-1}) =$$

$$= \left(\sum_{i=1}^{n-1} \Phi_i(y_i, y_{i-1}) + \Phi_n(y_n, y_{n-1}) \right)$$

Вычисление оптимальной разметки. Функция Q.

$$Q(0, v) = 0$$

$$Q(k, v) = \max_{y_1, \dots y_{k-1}} \left(\sum_{i=1}^{k-1} \Phi_i(y_i, y_{i-1}) + \Phi_k(v, y_{k-1}) \right) =$$

$$= \max_{y_1, \dots y_{k-1}} \left(\sum_{i=1}^{k-2} \Phi_i(y_i, y_{i-1}) + \Phi_{k-1}(y_{k-1}, y_{k-2}) + \Phi_k(v, y_{k-1}) \right) =$$

$$= \max_{y_{k-1}} \max_{y_1, \dots y_{k-2}} \left(\sum_{i=1}^{k-2} \Phi_i(y_i, y_{i-1}) + \Phi_{k-1}(y_{k-1}, y_{k-2}) + \Phi_k(v, y_{k-1}) \right) =$$

$$= \max_{y_{k-1}} \left(Q(k-1, y_{k-1}) + \Phi_k(v, y_{k-1}) \right)$$

Нахождение $\hat{y}$ по Q

$$\hat{y}_n = \arg\max_{y_n} \max_{y_1,\dots,y_{n-1}} a(x, y) = \arg\max_{y_n} Q(n, y_n)$$

$$\hat{y}_k = \arg\max_{y_k} \max_{y_1,\dots,y_{k-1}} \max_{y_{k+1},\dots,n} a(x, y) = \arg\max_{y_k} \max_{y_1,\dots,y_{k-1}} \Bigg($$

$$\sum_{i=1}^{k} \Phi_i(y_i, y_{i-1}) + \Phi_{k+1}(\hat{y}_{k+1}, y_k) + \sum_{i=k+2}^{n} \Phi_i(\hat{y}_i, \hat{y}_{i-1})\Bigg) =$$

$$= \arg\max_{y_k} \left(Q(k, y_k) + \Phi_{k+1}(\hat{y}_{k+1}, y_k)\right)$$

Алгоритм Витерби (получение оптимальной разметки)

Прямой ход — рекуррентное вычисление матрицы $Q \in \mathbb{R}^{n \times |Y|}$:

$$Q(0, v) = 0, \qquad Q(k, v) = \max_{y_{k-1} \in Y} \left(Q(k-1, y_{k-1}) + \Phi_k(v, y_{k-1}) \right)$$

Обратный ход — вычисление оптимальной разметки $\hat{y} \in Y^n$:

$$\hat{y}_n = \arg\max_{y_n \in Y} Q(n, y_n), \quad \hat{y}_k = \arg\max_{y_k \in Y} \left(Q(k, y_k) + \Phi_{k+1}(\hat{y}_{k+1}, y_k) \right)$$

Какая вычислительная сложность алгоритма?

[1] Viterbi. Error bounds for onvolutional odes and an asymptoti ally optimum de oding algorithm. 1967

Алгоритм Витерби (получение оптимальной разметки)

Прямой ход — рекуррентное вычисление матрицы $Q \in \mathbb{R}^{n \times |Y|}$:

$$Q(0, v) = 0, \qquad Q(k, v) = \max_{y_{k-1} \in Y} \left(Q(k-1, y_{k-1}) + \Phi_k(v, y_{k-1}) \right)$$

Обратный ход — вычисление оптимальной разметки $\hat{y} \in Y^n$:

$$\hat{y}_n = \arg\max_{y_n \in Y} Q(n, y_n), \quad \hat{y}_k = \arg\max_{y_k \in Y} \left(Q(k, y_k) + \Phi_{k+1}(\hat{y}_{k+1}, y_k) \right)$$

Какая вычислительная сложность алгоритма?

Прямой ход $O(n|Y|^2\hat{d})$, обратный ход $O(n|Y|^2)$, $\hat{d}$ — количество ненулевых признаков.

[1] Viterbi. Error bounds for onvolutional odes and an asymptoti ally optimum de oding algorithm. 1967

Вычисление градиента функционала

Градиент одного слагаемого log-правдоподобия по θ:

$$\frac{\partial}{\partial\theta_j}\ln p_\theta(y|x) = F_j(x,y) - \frac{\partial}{\partial\theta_j}\ln Z(x,\theta) =$$

$$= F_j(x,y) - \frac{1}{Z(x,\theta)}\frac{\partial}{\partial\theta_j}\sum_{y\in Y^n}\exp\left(\sum_{j=1}^{d}\theta_j F_j(x,y)\right) =$$

$$= F_j(x,y) - \frac{1}{Z(x,\theta)}\left(\sum_{y\in Y^n}\exp\left(\sum_{j=1}^{d}\theta_j F_j(x,y)\right)F_j(x,y)\right) =$$

$$= F_j(x,y) - \sum_{y\in Y^n}F_j(x,y)p_\theta(y|x)$$

Вычисление градиента функционала

Подставим в градиент выражение F_j через f_j:

$$\sum_{y \in Y^n} F_j(x, y) p_\theta(y|x) = \sum_{y \in Y^n} p_\theta(y|x) \sum_{i=1}^{n} f_j(y_i, y_{i-1}, x, i) =$$

$$= \sum_{i=1}^{n} \sum_{y_i \in Y} \sum_{y_{i-1} \in Y} p_\theta(y_{i-1}, y_i|x) f_j(y_i, y_{i-1}, x, i)$$

Мы избавимся от экспоненциальной сложности, если научимся эффективно вычислять $p_\theta(y_{i-1}, y_i|x)$.

Вычисление совместной вероятности двух меток

Разделим выражение внутри суммы на три множителя, первый и третий будут содержать элементы из одного суммирования:

$$p_\theta(y_{k-1}, y_k|x) = \frac{1}{Z(x,\theta)}\left(\sum_{y_1,\dots,y_{k-2}}\sum_{y_{k+1},\dots,y_n}\exp\left(\sum_{i=1}^{n}\Phi_k(y_i, y_{i-1})\right)\right) =$$

$$= \frac{1}{Z(x,\theta)}\Bigg(\sum_{y_1,\dots,y_{k-2}}\sum_{y_{k+1},\dots,y_n}\exp\left(\sum_{i=1}^{k-2}\Phi_i(y_i, y_{i-1}) + \Phi_{k-1}(y_{k-1}, y_{k-2})\right)\times$$

$$\times\exp(\Phi_k(y_k, y_{k-1}))\times\exp\left(\Phi_{k+1}(y_{k+1}, y_k) + \sum_{i=k+2}^{n}\Phi_i(y_i, y_{i-1}))\right)\Bigg) = (*)$$

Вычисление совместной вероятности двух меток

$$(*) = \frac{\exp(\Phi_k(y_k, y_{k-1}))}{Z(x,\theta)} \times$$

$$\times \left(\sum_{y_1,\dots,y_{k-2}} \exp\left(\sum_{i=1}^{k-2} \Phi_i(y_i, y_{i-1}) + \Phi_{k-1}(y_{k-1}, y_{k-2}) \right) \times \right.$$

$$\left. \times \sum_{y_{k+1},\dots,y_n} \exp\left(\Phi_{k+1}(y_{k+1}, y_k) + \sum_{i=k+2}^{n} \Phi_i(y_i, y_{i-1})) \right) \right) =$$

$$= \frac{\exp(\Phi_k(y_k, y_{k-1}))\alpha(k-1, y_{k-1})\beta(k, y_k)}{Z(x,\theta)}$$

Вектора «вперёд» и «назад» (forward and backward vectors)

$\alpha(k, v)$ — «вперёд» вектор, ненормированная вероятность начала последовательности:

$$\alpha(k, v) = \sum_{y_1, \dots, y_{k-1}} \exp\left(\sum_{i=1}^{k-1} \Phi_i(y_i, y_{i-1}) + \Phi_k(v, y_{k-1})\right)$$

$\beta(k, u)$ — «назад» вектор, ненормированная вероятность конца последовательности:

$$\beta(k, u) = \sum_{y_{k+1}, \dots, y_n} \exp\left(\Phi_{k+1}(y_{k+1}, u) + \sum_{i=k+2}^{n} \Phi_i(y_i, y_{i-1})\right)$$

Пересчёт векторов «вперёд»

Для векторов «вперёд» и «назад» можно вывести рекуррентные формулы аналогичные формулам в алгоритме Витерби:

$$\alpha(k, v) = \sum_{y_1,\dots,y_{k-1}} \exp\left(\sum_{i=1}^{k-1} \Phi_i(y_i, y_{i-1}) + \Phi_k(v, y_{k-1})\right) =$$

$$= \sum_{y_1,\dots,y_{k-2}} \sum_{y_{k-1}} \exp\left(\sum_{i=1}^{k-2} \Phi_i(y_i, y_{i-1}) + \Phi_{k-1}(y_{k-1}, y_{k-2})\right) \times$$

$$\times \exp\left(\Phi_k(v, y_{k-1})\right) = \sum_{y_{k-1}} \alpha(k-1, y_{k-1}) \exp\left(\Phi_k(v, y_{k-1})\right)$$

Пересчёт векторов «вперёд»

Пересчёт «вперёд» векторов

$$\alpha(0, v) = \mathbb{I}[v = \text{<START>}]$$

$$\alpha(k, v) = \sum_{u \in Y} \alpha(k-1, u) \exp\left(\Phi_k(v, u)\right)$$

Пересчёт «назад» векторов

$$\beta(n+1, u) = \mathbb{I}[u = \text{<END>}]$$

$$\beta(k, u) = \sum_{v \in Y} \beta(k+1, v) \exp\left(\Phi_{k+1}(v, u)\right)$$

Подрузамеваем, что в начале и конце последовательности стоят специальные метки.

Собираем всё вместе! Формула вычисления градиента

$$\frac{\partial}{\partial \theta_j} \ln p_\theta(y|x) = F_j(x,y) - \sum_{i=1}^{n} \sum_{y_i, y_{i-1} \in Y} p_\theta(y_{i-1}, y_i|x) f_j(y_i, y_{i-1}, x, i)$$

$$p_\theta(y_{i-1}, y_i|x) = \frac{\exp(\Phi_i(y_i, y_{i-1}))\alpha(i-1, y_{i-1})\beta(i, y_i)}{Z(x,\theta)}$$

$$Z(x,\theta) = \sum_{v \in Y} \alpha(n, v)$$

Какая вычислительная сложность алгоритма?

Собираем всё вместе! Формула вычисления градиента

$$\frac{\partial}{\partial\theta_j}\ln p_\theta(y|x) = F_j(x,y) - \sum_{i=1}^{n}\sum_{y_i,y_{i-1}\in Y} p_\theta(y_{i-1},y_i|x) f_j(y_i,y_{i-1},x,i)$$

$$p_\theta(y_{i-1},y_i|x) = \frac{\exp(\Phi_i(y_i,y_{i-1}))\alpha(i-1,y_{i-1})\beta(i,y_i)}{Z(x,\theta)}$$

$$Z(x,\theta) = \sum_{v\in Y}\alpha(n,v)$$

Какая вычислительная сложность алгоритма?

$O(\hat{d}n|Y|^2)$

Резюме по linear-CRF

- Linear-CRF — аналог мультиномиальной регрессии для последовательности
- Коэффициенты Linear-CRF можно обучить при помощи SGD
- Градиент модели вычисляется при помощи алгоритма forward-backward за $O(\hat{d}n|Y|^2)$
- Процедура получения оптимальной разметки готовой моделью производится при помощи алгоритма Витерби за $O(n|Y|^2\hat{d})$

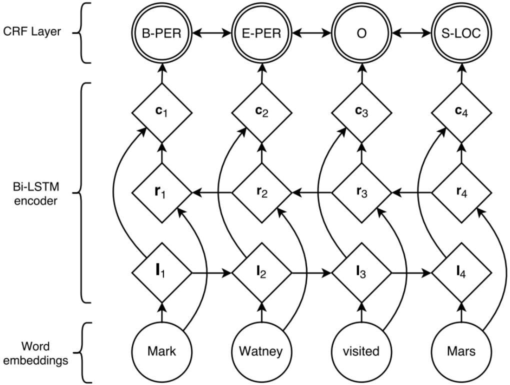
Linear-CRF как слой

Linear-CRF — дифференцируемая функция вычисления вероятности последовательности и может использоваться как слой в нейросети.

Зачем это может быть надо?

- Согласование предсказаний для соседних меток
- Улучшение качества модели (при недостатке обучающих данных)

Linear-CRF как слой



- CRF используется в конце сети вместо $\mathrm{softmax}$
- На вход CRF поступает последовательность $(x_1, \ldots, x_n)$, $x_i \in \mathbb{R}^{|Y|}$
- Таким образом, CRF — обучаемая постобработка модели

Признаки CRF для постобработки выходов сети

Выход, соответствующий метке v элемента x_i обозначим $x_i(v)$.
Будем использовать две группы разреженных признаков:

- $|Y| \times |Y|$ «унарных» признаков

$$\phi_{uv}(y_{i-1}, y_i, x_i) = \mathbb{I}[y_{i-1} = u]\mathbb{I}[y_i = v]x_i(v)$$

- $|Y| \times |Y|$ «бинарных» признаков

$$\psi_{uv}(y_{i-1}, y_i) = \mathbb{I}[y_{i-1} = u]\mathbb{I}[y_i = v]q(u, v)$$

$q(u, v)$ — любая попарная статистика меток u и v или 1

Преобразование выражения для признаков

Линейные коэффициенты модели обозначим $\theta(u, v)$ и $w(u, v)$.

$$\Phi_{x,i}(y_{i-1}, y_i) =$$
$$= \sum_{u\in Y}\sum_{v\in Y}\left(\theta(u, v)\phi_{uv}(y_{i-1}, y_i, x_i) + w(u, v)\psi_{uv}(y_{i-1}, y_i)\right) =$$
$$= \sum_{u\in Y}\sum_{v\in Y}\mathbb{I}[y_{i-1} = u]\mathbb{I}[y_i = v](\theta(u, v)x_i(v) + w(u, v)q(u, v)) =$$
$$= \theta(y_{i-1}, y_i)x_i(y_i) + w(y_{i-1}, y_i)q(y_{i-1}, y_i)$$

Часто при реализации полагают $\theta(u, v) = 1 \quad \forall u \in Y, v \in Y$.

Улучшение от CRF в biLSTM[1]

Добавление CRF слоя улучшает результаты:

Table 2: Comparison of tagging performance on POS, chunking and NER tasks for various models.

		POS	CoNLL2000	CoNLL2003
Random	Conv-CRF (Collobert et al., 2011)	96.37	90.33	81.47
	LSTM	97.10	92.88	79.82
	BI-LSTM	97.30	93.64	81.11
	CRF	97.30	93.69	83.02
	LSTM-CRF	**97.45**	93.80	84.10
	BI-LSTM-CRF	97.43	**94.13**	**84.26**
Senna	Conv-CRF (Collobert et al., 2011)	97.29	94.32	88.67 (89.59)
	LSTM	97.29	92.99	83.74
	BI-LSTM	97.40	93.92	85.17
	CRF	97.45	93.83	86.13
	LSTM-CRF	97.54	94.27	88.36
	BI-LSTM-CRF	**97.55**	**94.46**	**88.83 (90.10)**

[1] Huang et al (2015); Bidirectional LSTM-CRF Models for Sequence Tagging.

Добавление CRF слоя на практике

- Если у вас мало обучающих данных, может сильно помочь
- Есть реализация в Pytorch (ссылка)
- Сильно замедляет модель при большом количестве уникальных меток
- Можно не обучать коэффициенты модели
- Можно использовать более простые эвристики