Математические методы анализа текстов

Семинар 2

Языковые модели. Статистический машинный перевод, задача выравнивания.

Мурат Апишев (great-mel@yandex.ru) [1]

МГУ им. М. В. Ломоносова

15 февраля, 2018

[1] Подготовлена с использованием материалов Анны Потапенко и Дэвида Тэлбота

Содержание занятия

- Языковые модели (повторение)
  - Определение
  - Приложения
  - Виды сглаживания
- EM-алгоритм (повторение)
  - Интуиция
  - Общий вид
  - Приложения
- Статистический машинный перевод
  - Модели языка и перевода, задача выравнивания
  - Модели IBM, HMM
  - Модели выравнивания предложений

Языковая модель

- Хотим присвоить вероятности последовательностям слов
- *Языковой моделью* называется модель, умеющая вычислять хоть одну из этих вероятностей:
  1. $P(W) = P(w_1, \ldots, w_n)$
  2. $P(w_n \mid w_1, \ldots, w_{n-1})$
- Цепное правило:

$$P(X_1, \ldots, X_n) = P(X_1)P(X_2 \mid X_1) \ldots P(X_n \mid X_1, \ldots, X_{n-1})$$

- Формула условной вероятности:

$$P(X_n \mid X_1, \ldots, X_{n-1}) = \frac{P(X_1, \ldots, X_n)}{P(X_1, \ldots, X_{n-1})},$$

Языковая модель

- Для оценивания $P(X_n \mid X_1, \ldots, X_{n-1})$ нужно посчитать $P(X_1, \ldots, X_n)$ и $P(X_1, \ldots, X_{n-1})$
- С ростом n число всевозможных последовательностей растёт экспоненциально $\Rightarrow$

  Проблема: для большого n эти вероятности близки к нулю
- Для упрощения применим *марковское предположение*:

$$P(X_n \mid X_1, \ldots, X_{n-1}) \approx P(X_n \mid X_{n-k+1}, \ldots, X_{n-1}), \quad k \ll n$$

- Окончательная формула ($w_{i-k}^{i} := w_{i-k}, \ldots, w_i$)

$$P(w_1, \ldots, w_n) = \prod_i P(w_i \mid w_{i-k+1}^{i-1})$$

Приложения

- Генерация текста
- Распознавание речи/текста
- Машинный перевод
- Исправление опечаток
- Определения языка
- Определение части речи (POS)
- ...

Открытые вопросы

1. Как обучать вероятности?
2. Как использовать их для генерации текста?
3. Какие проблемы могут возникнуть при обучении?
4. Как эти проблемы решать?

Ответы

1. Статистически по обучающему корпусу:

$$\hat{P}_S(w_N \mid w_1^{N-1}) = \frac{c(w_1^N)}{c(w_1^{N-1})},$$

   $c(w_1^N)$ — это число последовательностей $w_1, \ldots, w_N$ в корпусе

2. Сэмплируем из полученных эмпирических распределений.

3. Для многих вероятностей даже при небольших n значения могут быть нулевыми.

4. Увеличение обучающего корпуса, сглаживание частот, откат.

Борьба с нулями

- *Add-one smoothing* (сглаживание Лапласа):

$$\hat{P}_{AOS}(w_N \mid w_1^{N-1}) = \frac{c(w_1^N) + \delta}{c(w_1^{N-1}) + \delta V},$$

V — это размер словаря, а δ — некоторая фиксированная константа.

Чем плох такой подход?

- *Katz smoothing* (простой откат): если не получается применить модель высокого порядка, пробуем для данного слова модель меньшего порядка с понижающим множителем.

Получим не вероятностное распределение!

Борьба с нулями

- *Jelinek-Mercer smoothing* (интерполяционное сглаживание): заведем вектор $\bar{\lambda} = (\lambda_1, \ldots, \lambda_N)$, такой, что $\sum_i \lambda_i = 1$ и $\lambda_i \geqslant 0$. Тогда

$$\hat{P}_{IS}(w_N \mid w_1^{N-1}) = \sum_{i=1}^{N} \lambda_i \hat{P}_S(w_N \mid w_{N-i+1}^{N-1}).$$

- Другие виды сглаживаний:
  - *Good-Turing estimate*
  - *Witten-Bell smoothing*
  - *Absolute discounting*
  - *Kneser-Ney smoothing*

Пример: генерация текстов Шекспира

- **Униграммная модель:**

  To him swallowed confess hear both. Which. Of save on trail for are ay device and rote life have. Every enter now severally so, let. Hill he late speaks; or! a more to leg less first you enter.

- **Биграммная модель:**

  What means, sir. I confess she? then all sorts, he is trim, captain. Why dost stand forth thy canopy, forsooth; he is this palpable hit the King Henry. Live king. Follow. What we, hath got so she that I rest and sent to scold and nature bankrupt, nor the first gentleman?

Пример: генерация текстов Шекспира

- **3-граммная модель:**

  Sweet prince, Falstaff shall die. Harry of Monmouth's grave. This shall forbid it should be branded, if renown made it empty. What is't that cried? Indeed the duke; and had a very good friend. Fly, and will rid me these news of price. Therefore the sadness of parting, as they say, 'tis done.

- **4-граммная модель:**

  King Henry. What! I will go seek the traitor Gloucester. Exeunt some of the watch. A great banquet serv'd in; Will you not tell me who I am? It cannot be but so. Indeed the short and the long. Marry, 'tis a noble Lepidus. They say all lovers swear more performance than they are wont to keep obliged faith unforfeited.

EM-алгоритм: эксперимент

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 монеток

Хотим узнать:

- $\lambda = p(\text{синяя}) =$
- $\theta_1 = p(\text{орел}|\text{синяя}) =$
- $\theta_2 = p(\text{орел}|\text{красная}) =$

EM-алгоритм: эксперимент

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 монеток

Хотим узнать:

- $\lambda = p(\text{синяя}) = \frac{\text{синий бросок}}{N}$
- $\theta_1 = p(\text{орел}|\text{синяя}) = \frac{(\text{орел,синяя})}{\text{синяя}}$
- $\theta_2 = p(\text{орел}|\text{красная}) = \frac{(\text{орел,красная})}{\text{красная}}$

EM-алгоритм: эксперимент

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 монеток

Метод максимума правдоподобия:

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

EM-алгоритм: эксперимент

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 монеток

Хотим узнать:

- $\lambda = p(\text{синяя}) =$
- $\theta_1 = p(\text{орел}|\text{синяя}) =$
- $\theta_2 = p(\text{орел}|\text{красная}) =$

EM-алгоритм: эксперимент

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 монеток

Метод максимума правдоподобия:

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

EM-алгоритм в общем виде

X — наблюдаемые переменные; T — скрытые; Θ — параметры.

Задача максимизации *неполного* правдоподобия:

$$\ln p(X|\Theta) = \ln \int p(X, T|\Theta) dT \to \max_{\Theta}$$

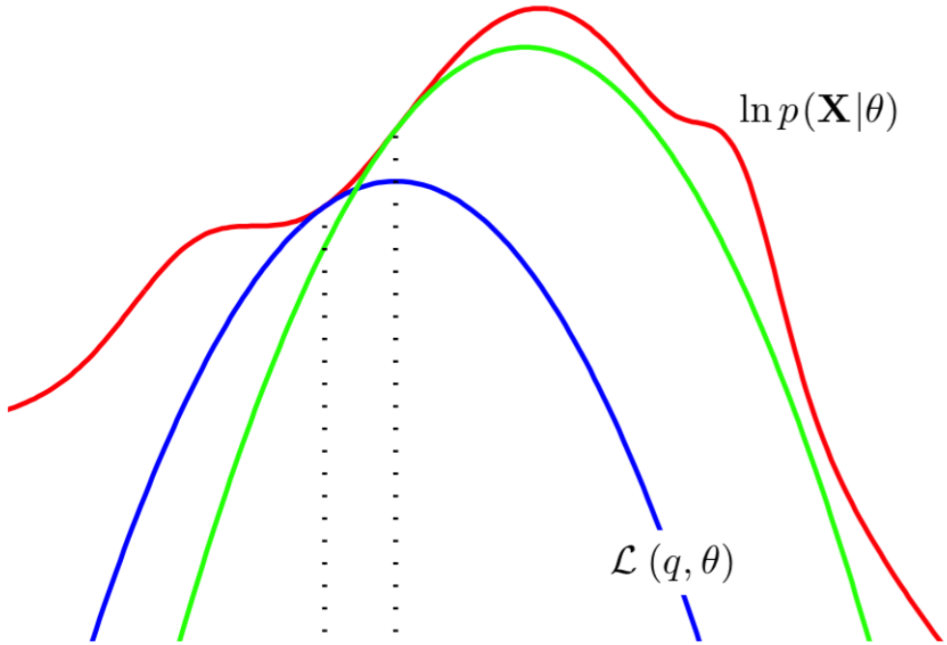
Преобразуем:

$$\ln p(X|\Theta) = \int \ln p(X|\Theta) q(T) dT = \int q(T) \ln \frac{p(X, T|\Theta)}{p(T|X, \Theta)} \frac{q(T)}{q(T)} dT$$

$$= \underbrace{\int q(T) \ln \frac{p(X, T|\Theta)}{q(T)} dT}_{L\{q,\Theta\}} + \underbrace{\int q(T) \ln \frac{q(T)}{p(T|X, \Theta)} dT}_{KL(q(T)||p(T|X,\Theta))}$$

- **E-step:** $KL(q(T)||p(T|X, \Theta)) \to \min\limits_{q(T)} \Leftrightarrow q(T) = p(T|X, \Theta)$
- **M-step:** $L\{q, \Theta\} \to \max\limits_{\Theta} \Leftrightarrow \mathbb{E}_{q(T)} \ln p(X, T|\Theta) \to \max\limits_{\Theta}$

Геометрическая интерпретация оптимизационного процесса



- **E-step:** $KL(q(T)||p(T|X,\Theta)) \to \min\limits_{q(T)} \Leftrightarrow q(T) = p(T|X,\Theta)$
- **M-step:** $L\{q,\Theta\} \to \max\limits_{\Theta} \Leftrightarrow \mathbb{E}_{q(T)} \ln p(X,T|\Theta) \to \max\limits_{\Theta}$

Пример: модель смеси распределений

Наблюдаемые переменные:
$X = x_1, x_2, \ldots x_N$ – выборка из смеси распределений

Скрытые переменные:
$T = t_1, t_2, \ldots t_N$ – номера компонент смеси

Параметры:
$\Theta = \theta_1, \ldots \theta_k, w_1, \ldots w_k$

Модель: K компонент, каждая имеет свое распределение:

$$p(x_i|t_i = j, \Theta) = p_j(x_i|\theta_j)$$

Компоненты выбираются с весами:

$$p(t_i = j|\Theta) = w_j;$$

Если бы видели скрытые переменные...

Максимизировали бы полное правдоподобие:

$$\ln p(X, T|\Theta) = \sum_{i=1}^{m} \ln p(x_i, t_i|\Theta) = \sum_{i=1}^{m} \sum_{j=1}^{k} [t_i = j] \ln w_j p_j(x_i|\theta_j) \to \max_{\Theta}$$

при ограничениях $\sum_{j=1}^{k} w_j = 1; \;\; w_j \geqslant 0.$

Оценки:

$$\sum_{i=1}^{m} [t_i = j] \ln p_j(x_i|\theta_j) \to \max_{\theta_j}; \qquad w_j = \frac{1}{m} \sum_{i=1}^{m} [t_i = j]$$

. . . но мы их не видим

EM-алгоритм для максимизации неполного правдоподобия:

$$\ln p(X|\Theta) = \sum_{i=1}^{N} \ln p(x_i|\Theta) = \sum_{i=1}^{N} \ln \sum_{j=1}^{k} w_j p_j(x_i|\theta_j) \to \max_{\Theta}$$

- **E-шаг:** оцениваем апостериорные распределения на скрытые переменные по формуле Байеса:

$$p(t_i = j|x_i, \Theta) = \frac{p(t_i = j|\Theta)p(x_i|t_i = j, \Theta)}{p(x_i|\Theta)} = \frac{w_j p_j(x_i|\theta_j)}{\sum_{l=1}^{k} w_l p_l(x_i|\theta_l)}$$

- **M-шаг:** максимизируем м.о. полного правдоподобия:

$$\mathbb{E}_{q(T)} \ln p(X, T|\Theta) = \mathbb{E}_{q(T)} \sum_{i=1}^{m} \ln p(x_i, t_i|\Theta) =$$

$$= \sum_{i=1}^{m} \sum_{j=1}^{k} p(t_i = j|x_i, \Theta) \ln w_j p_j(x_i|\theta_j) \to \max_{\Theta}$$

при ограничениях $\sum_{j=1}^{k} w_j = 1;\ \ w_j \geqslant 0.$

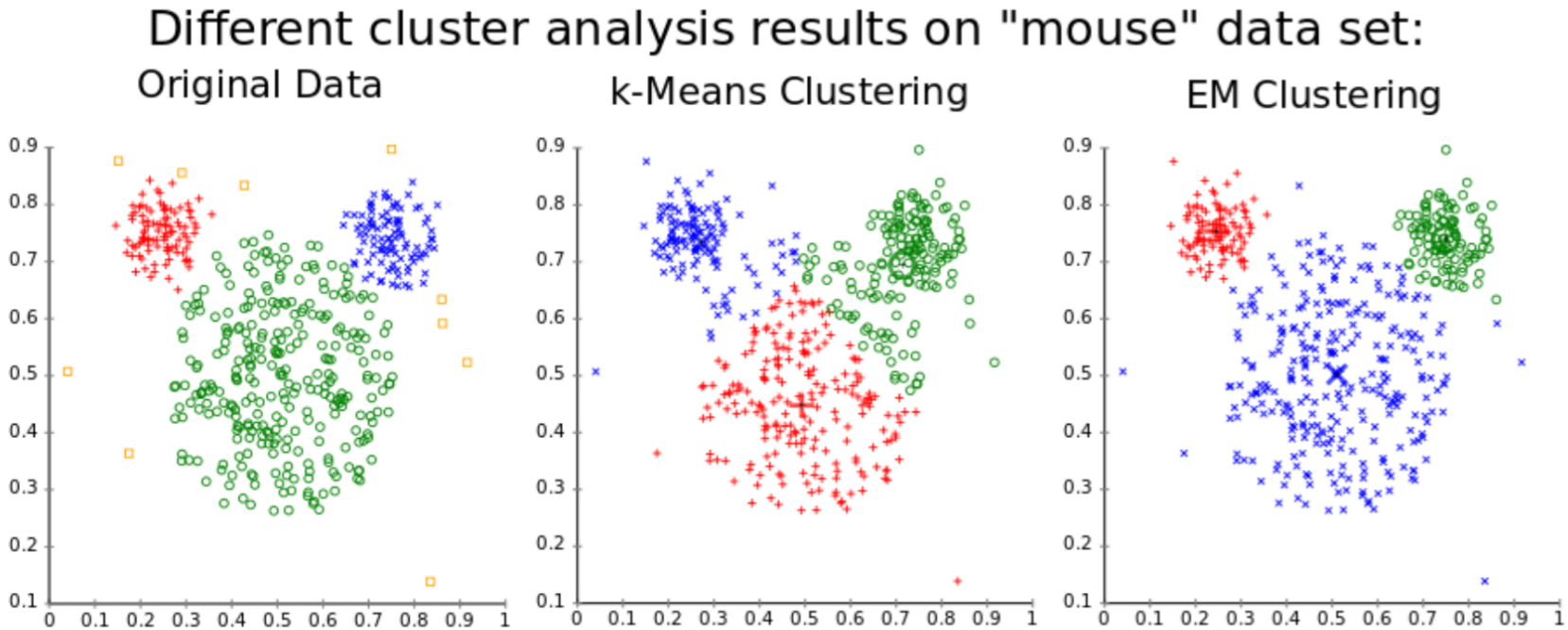
Оценки:

$$\sum_{i=1}^{m} p(t_i = j|x_i, \Theta) \ln p_j(x_i|\theta_j) \to \max_{\theta_j}; \qquad w_j = \frac{1}{m} \sum_{i=1}^{m} p(t_i = j|x_i, \Theta)$$

Приложения EM-алгоритма

- Кластеризация (мягкая, в отличе от k-means)
- Сегментация изображений (месь гауссиан)
- Машинный перевод (скрытые переменные — выравнивания)
- Определение частей речи (алгоритм для HMM)
- Тематическое моделирование (скрытые переменные — тематики слов в тексте)
- . . . Везде, где есть скрытые переменные и неполное правдоподобие

Кластеризация: EM-алгоритм vs. k-means



k-means:

- Отнести каждую точку к кластеру с ближайшим центром
- Пересчитать центр каждого кластера, усреднив его точки

Кластеризация: EM-алгоритм vs. k-means

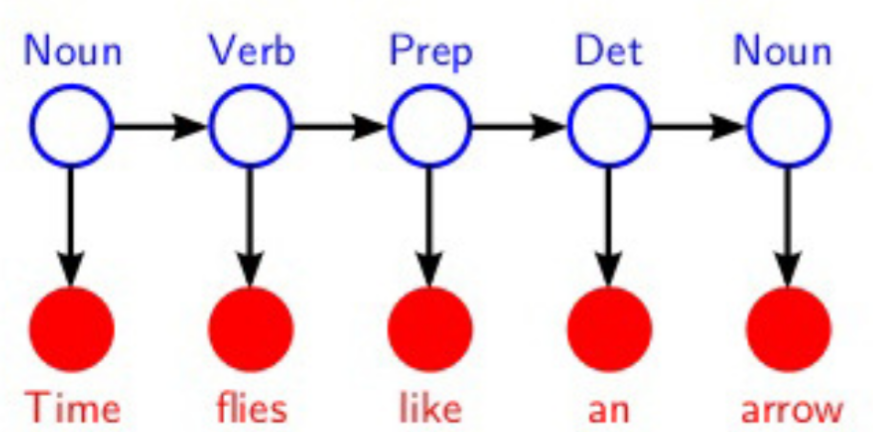
EM для смеси гауссиан	**k-means**
$\sum_{i=1}^{N} \ln w_j \mathcal{N}(x_i \mid \mu_j, \Sigma_j) \to \max\limits_{\mu_j, \Sigma_j, w_j}$	$\sum_{i=1}^{N} \sum_{j=1}^{K} r_{ij} \|\|x_i - \mu_j\|\|^2 \to \min\limits_{r_{ij}, \mu_j}$
E-шаг: вероятности $p(t_i = j)$	присваивания $r_{ij} \in \{0, 1\}$
M-шаг: оценки на μ_j и Σ_j	оценки только на центры μ_j

Если фиксировать матрицы ковариаций $\Sigma = \varepsilon I$ и устремить $\varepsilon \to 0$, то EM превращается в k-means.

GMM для модели цветов объектов на изображении

- Гауссовская смесь GMM — Gaussian Mixture Model
- Каждая компонента описывает распределение в пространстве RGB для пикселей определенного объекта
- Может использоваться для сегментации изображения

Распознавание частей речи (POS-tagging): HMM



Наблюдаемые переменные: слова

Скрытые переменные: части речи

Параметры модели: $p(t_{i+1}|t_i)$ и $p(x_i|t_i)$

- **E-шаг:** делаем предположения о частях речи
- **M-шаг:** учим параметры, максимизируя вероятность последовательности слов в сделанных предположениях

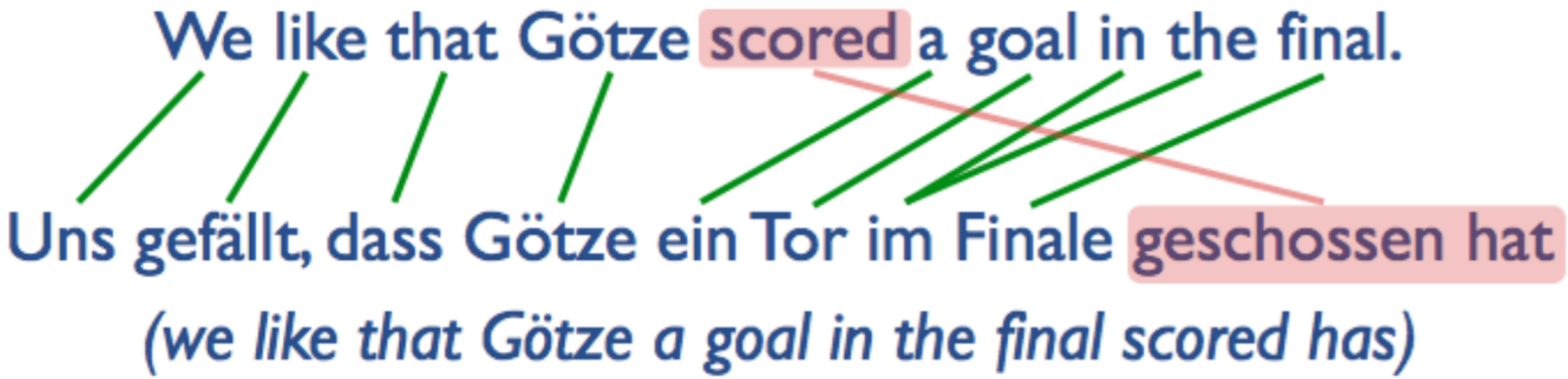
Статистический машинный перевод

e, f – предложения на двух разных языках

Предложения должны быть адекватны с точки зрения языка и с точки зрения перевода:

$$\hat{e} = \arg\max_e \underbrace{p(e)}_{language\_model} \underbrace{p(f|e)}_{translation\_model}$$

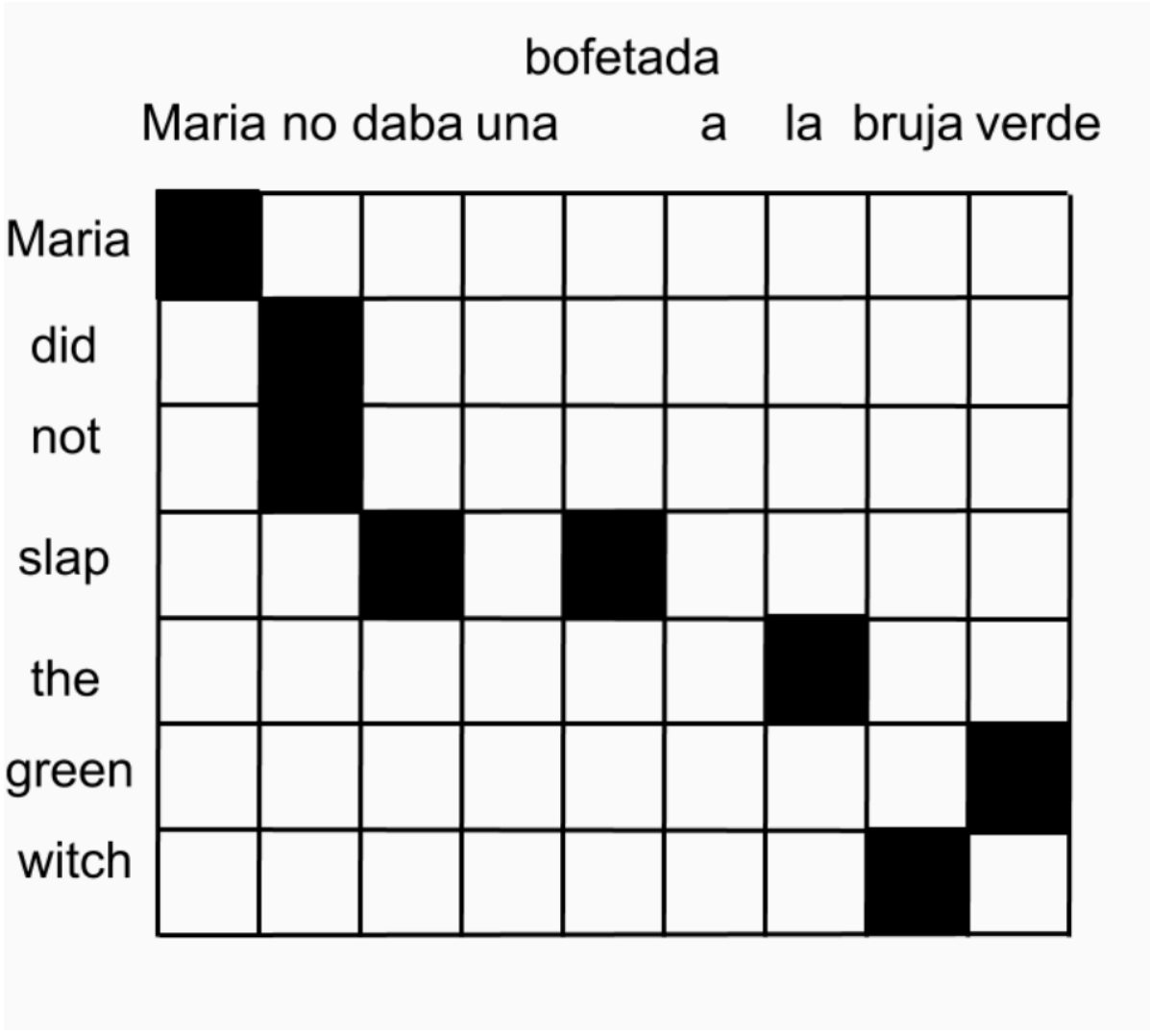
Перевод зависит от выравнивания:



Языковая модель

- Языковая модель $p(e)$ позволяет нам среди всех возможных переводов e выбрать тот, который более специфичен для целевого языка.
- Присваиваем высокие вероятности адекватным кандидатам:
  - The cat in the hat.
  - Green eggs and ham.
- И низкие — маловероятным:
  - Cat the hat in the.
  - Eggs ham green and.

Модель перевода: матрица выравнивания



Модель перевода: обозначения и правдоподобие

- D — корпус предложений $\{(e, f)_1, \ldots, (e, f)_m\}$
- e — предложение длины I, f — предложение длины J
- θ — параметры модели перевода
- $a_k \in \mathcal{A}^{I \times J}$ — матрица выравнивания предложений k

Неполное правдоподобие модели:

$$p(D \mid \theta) = \prod_{k \in D} p(f_k \mid e_k, \theta) = \prod_{k \in D} \sum_{a_k \in \mathcal{A}} p(a_k, f_k \mid e_k, \theta) =$$

$$= \prod_{k \in D} \sum_{a_k \in \mathcal{A}} \underbrace{p(a_k \mid e_k, \theta)}_{\text{prior}} \underbrace{p(f_k \mid e_k, a_k, \theta)}_{\text{translation model}}$$

IBM Model 1

Введём следующие общие предположения:

1. Будем слова в предложениях сопоставлять одно к одному $\Rightarrow$ матрица выравнивания превращается в вектор
2. Каждое слово генерируется независимо от контекста (но зависимость между элементами a сохраняется)

Тогда правдоподобие пары предложений можно записать так:

$$p(f \mid e) = \sum_{a \in \mathcal{A}} p(a \mid e, \theta) p(f \mid e, a, \theta) \approx$$

$$\approx \prod_{j=1}^{J} \sum_{a \backslash j \in \mathcal{A}} p(a \backslash j \mid e, \theta) \sum_{a_j=1}^{I} p(a_j \mid a \backslash j, e, \theta) p(f_j \mid e_{a_j}, \theta)$$

IBM Model 1

Дополнительное предположение:

Различные присваивания независимы друг от друга $\Rightarrow$ $p(a_j \mid a \backslash j, e, \theta) = p(a_j \mid e, \theta) \Rightarrow$ сумма по $a \backslash j$ уходит вправо и обращается в единицу

Тогда правдоподобие пары предложений можно записать так:

$$p(f \mid e) \approx \prod_{j=1}^{J} \sum_{a_j=1}^{I} p(a_j \mid, e, \theta) p(f_j \mid e_{a_j}, \theta) \sum_{a \backslash j \in \mathcal{A}} p(a \backslash j \mid e, \theta) =$$

$$= \prod_{j=1}^{J} \sum_{a_j=1}^{I} p(a_j \mid e, \theta) p(f_j \mid e_{a_j}, \theta)$$

IBM Model 1

Предположение о независимости позволяет иметь на пару слов один параметр $t(f \mid e)$ — вероятность генерации слова f по сопоставленному ему слову e.

Основное предположение Model IBM 1: вероятности всех присваиваний равны, $p(a_j \mid e, \theta) = \varepsilon$.

Это позволяет записать м. о. правдоподобия для пары предложений:

$$\mathbb{E}[\log(f, a \mid e)] = \prod_{j=1}^{J} \sum_{a_j=1}^{I} p(a_j \mid f, e, \theta) \log p(f_j, a_j \mid e, \theta) =$$

$$= \prod_{j=1}^{J} \sum_{a_j=1}^{I} p(a_j \mid f, e, \theta) \log t(f_j \mid_{a_j}) \varepsilon$$

EM-алгоритм для Model IBM 1

- **E-шаг:** считаем апостериорное распределение на присваивания (формула Байеса):

$$p(a_j = i \mid f, e, \theta) = \frac{p(a_j = i \mid e, \theta) p(f_j \mid a_j = i, e, \theta)}{\sum_{i'=1}^{I} p(a_j = i' \mid e, \theta) p(f_j \mid a_j = i', e, \theta)}$$

- **M-шаг:** имея распределения на присваивания и корпус, пересчитаем значения параметров:

$$t(f \mid e)_{\text{new}} = \frac{\sum_{k \in D} \sum_j \sum_i p(a_{jk} = i \mid f, e, \theta) [f_{jk} = f \wedge e_{ik} = e]}{\sum_{k \in D} \sum_j \sum_i p(a_{jk} = i \mid f, e, \theta) [e_{ik} = e]}$$

Порождающий процесс Model IBM 1

Для каждого позиции j в f:

1. Выбираем $a_j \in [0, I]$ для $j-$го слова с вероятностью ε
2. Генерируем f_j с вероятностью $t(f_j \mid e_{a_j})$

Вероятность выровненного целевого предложения:

$$p(f, a \mid e) = \prod_j t(f_j \mid e_{a_j})$$

Маргинальная вероятность целевого предложения:

$$p(f \mid e) = \sum_{a_1=1}^{I} \cdots \sum_{a_J=1}^{I} \prod_j t(f_j \mid e_{a_j}) = \prod_{j=1}^{J} \sum_{a_j=1}^{I} t(f_j \mid e_{a_j})$$

Model IBM 2

- Выбираем $a_j = i$ не из равномерного распределения, а на основании абсолютных позиций i и j.
- Выбираем $f_j = f$ так же только на основании e_{a_i}.

Параметры: $d(a_j \mid j, I, J), \quad t(f_j \mid e_{a_j})$

Параметров стало намного больше!

Маргинальная вероятность целевого предложения:

$$p(f \mid e) = \prod_{j=1}^{J} \sum_{a_j=1}^{I} d(a_j \mid j, I, J) t(f_j \mid e_{a_j})$$

Hidden Markov Model

- Выбираем a_j на основании его расстояния от a_{j-1}.
- Выбираем $f_j = f$ так же только на основании e_{a_i}.

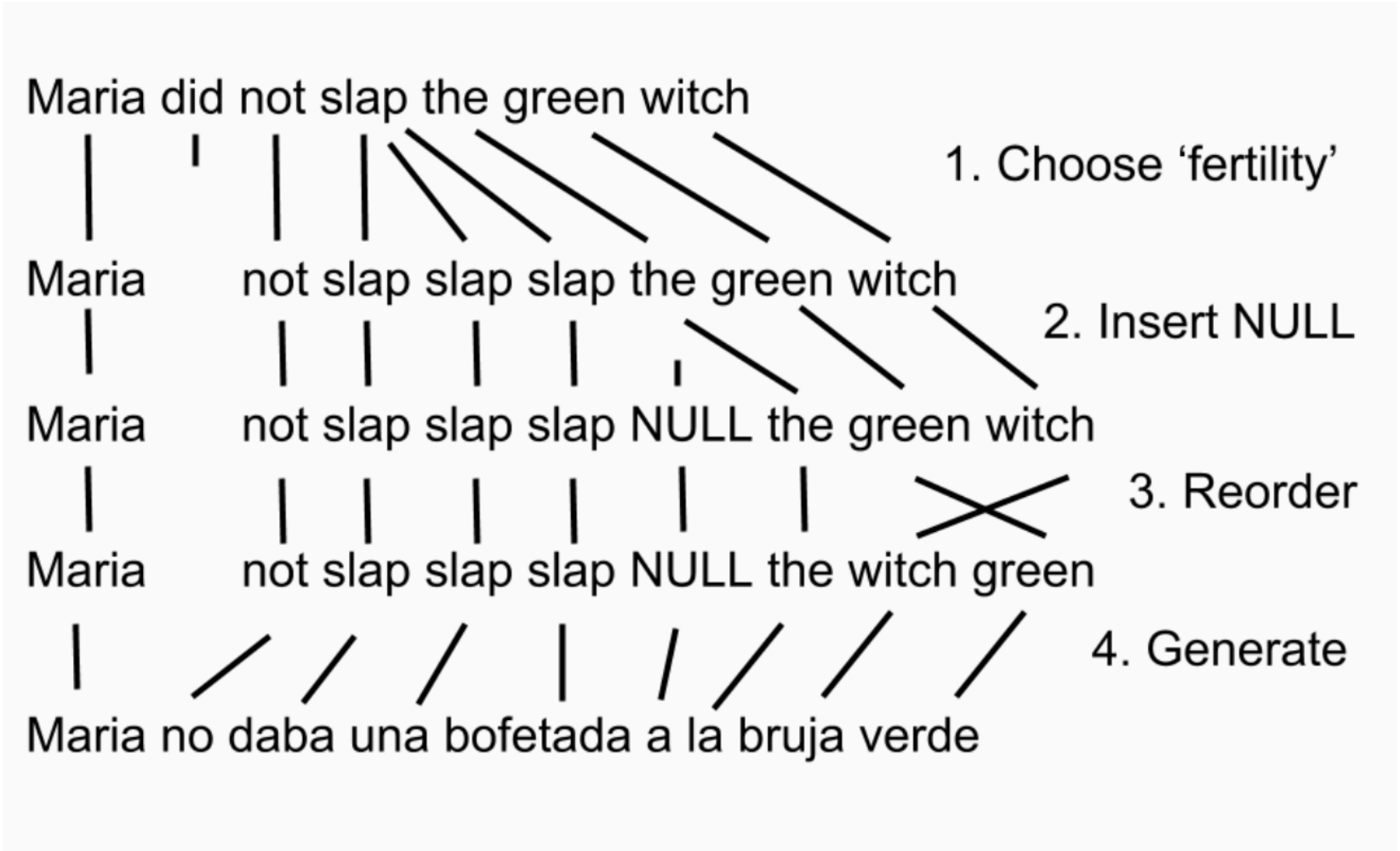
Параметры: $d(a_j \mid a_{j-1}, I, J), \quad t(f_j \mid e_{a_j})$

Вероятность выровненного целевого предложения:

$$p(f, a \mid e) = \prod_j d(a_j = 1 \mid a_{j-1}, I, J) t(f_j \mid e_{a_j})$$

Маргинальная вероятность целевого предложения может быть вычислена с помощью динамического программирования.

Model IBM 3


Maria did not slap the green witch
1. Choose ‘fertility’
Maria not slap slap slap the green witch
2. Insert NULL
Maria not slap slap slap NULL the green witch
3. Reorder
Maria not slap slap slap NULL the witch green
4. Generate
Maria no daba una bofetada a la bruja verde

Проблемы «word based» моделей

- Word based модели всё ещё часто используются для решения задачи выравнивания
- Менее успешны в задаче перевода

Проблемы:

- Теряется много зависимостей
- Параметры перевода не зависят от контекста
- Для моделей IBM 3+ переупорядочивание слов слабо зависит от самих слов
- Генерация целевого предложения требует большого количества шагов

Для обхода этих ограничений используются phrase based модели (будет в следующий раз).