

# Многомодальные тематические модели на гиперграфах

Жариков Илья Николаевич

`zharikov.i.n@yandex.ru`

Московский физико-технический институт  
Факультет управления и прикладной математики  
Кафедра интеллектуальных систем

Научный руководитель: д.ф.-м.н. К. В. Воронцов

Июнь, 2018

# Цель исследования

## Задача

Построение тематической модели, нахождение тематических векторов (профилей) объектов, выявление множества латентных тем.

## Проблема

Большинство существующих тематических моделей описывают только попарные взаимодействия между объектами разных типов (модальностей).

## Предлагается

Обобщить методы тематического моделирования на случай, когда исходные взаимодействия объектов не распадаются на попарные.

## Гиперграф

Примеры  
транзакций

- Данные социальной сети  
 $(u, w, d)$  — пользователь  $u$  написал слово  $w$  в блоге  $d$
- Данные сети интернет-рекламы  
 $(u, b, d)$  — пользователь  $u$  кликнул баннер  $b$  на странице  $d$
- Данные рекомендательной системы  
 $(u, f, s)$  — пользователь  $u$  оценил фильм  $f$  в ситуации  $s$
- Данные финансовых организаций  
 $(b, s, g)$  — покупатель  $b$  купил товар  $g$  у продавца  $s$

Транзакция  $\leftrightarrow$  определенное ребро.

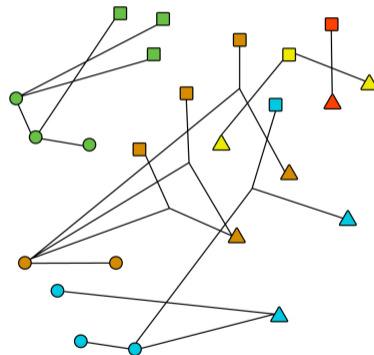
Данные  $\leftrightarrow$  ребра гиперграфа.

**Гиперграф** — обобщение графа, в котором ребром могут соединяться не только две вершины, но и **любое подмножество вершин**.

## Транзакционные данные — наблюдаемые ребра гиперграфа

Множество модальностей  $M$ :Множество типов рёбер  $K$ :

$d$					
$x$					
$n_{dx}$	3	3	4	2	4

 $n_{dx}$  — число рёбер  $(d, x)$  в гиперграфеМножество тем  $T$ :Хотим найти тематические вектора **всех** вершин

## Постановка задачи

Для оптимизации параметров применим принцип максимума правдоподобия:

$$\sum_{k \in K} \tau_k \sum_{(d,x) \in E_k} n_{dx} \ln \sum_{t \in T} \theta_{td} \prod_{v \in X} \varphi_{vtk} + \underbrace{R(\Phi, \Theta)}_{\text{регуляризатор}} \rightarrow \max_{\Phi, \Theta}$$

где  $\tau_k$  — вес ребёр типа  $k$ ,  $\theta_{td} = p(t | d)$ ,  $\varphi_{vtk} = p_k(v | t)$ .

Ограничения:

$$\sum_{v \in V_m} \varphi_{vtk} = 1, \varphi_{vtk} \geq 0$$

$$\sum_{t \in T} \theta_{td} = 1, \theta_{td} \geq 0$$

$d$  — документ

$v$  — вершина

$m$  — модальность

$k$  — тип ребра

$t$  — тема

## EM алгоритм

**E step.** Вычисление распределения тем для каждого  $(d, x)$ :

$$p_{tdx} = \operatorname{norm}_{t \in T} \left( \theta_{td} \prod_{v \in x} \varphi_{vtk} \right), \text{ где } \operatorname{norm}_{i \in I} a_i = \frac{\max\{a_i, 0\}}{\sum_{j \in I} \max\{a_j, 0\}}.$$

**M step.** Оценивание параметров модели:

$$\varphi_{vtk} = \operatorname{norm}_{v \in V_m} \left( n_{vtk} + \varphi_{vtk} \frac{\partial R}{\partial \varphi_{vtk}} \right); \quad n_{vtk} = \sum_{(d,x) \in E_k} [v \in x] \tau_k n_{dx} p_{tdx};$$

$$\theta_{td} = \operatorname{norm}_{t \in T} \left( n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}} \right); \quad n_{td} = \sum_{k \in K} \sum_{(d,x) \in E_k} \tau_k n_{dx} p_{tdx};$$

## Теорема

Если функция  $R(\Phi, \Theta)$  непрерывно дифференцируема и  $(\Phi, \Theta)$  — точка локального максимума рассматриваемой задачи, то выполняется система уравнений, описанная выше, относительно параметров  $\varphi_{vtk}$ ,  $\theta_{td}$  и вспомогательных переменных  $p_{tdx}$ ,  $n_{td}$  и  $n_{vtk}$ .

## Модельные данные:

- 1 Генерация матриц  $\Theta = p(t | d)$  и  $\Phi_k = p_k(v | t)$  для всех  $k \in K$ .

Число тем: 50

Число классов: 5

Число документов: 5000

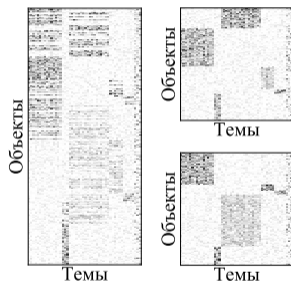
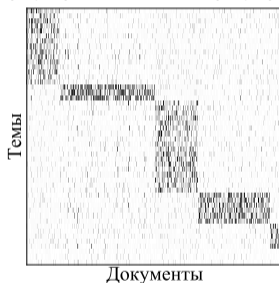
Число объектов: 1000

Число типов ребер: 9

Число модальностей: 3

$$\Theta \in \mathbb{R}^{50 \times 5000}$$

$$\Phi_k \in \mathbb{R}^{1000 \times 50}$$



- 2 Генерация данных (транзакций) на основе матриц  $\Theta$ ,  $\Phi_k$ .  
Общее число транзакций:  $\sim 13\,500\,000$ .

## Цели эксперимента:

- 1 Проверить, способен ли алгоритм восстановить параметры модели, с помощью которой были порождены данные.
- 2 Оценить устойчивость модели относительно инициализации и выбора числа тем.

## Постановка эксперимента:

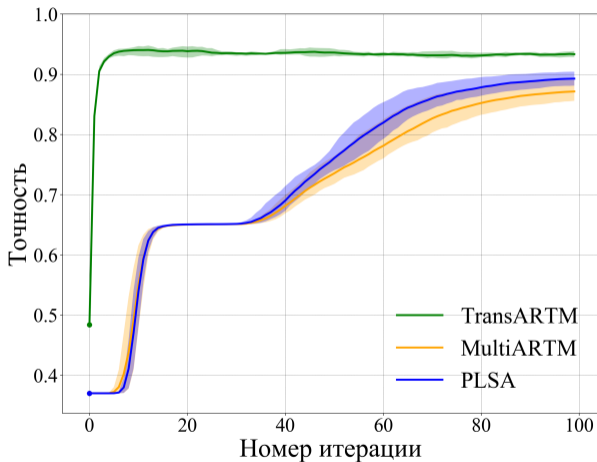
- 1 Вычисление матрицы  $\Theta$  с использованием различных методов:
  - PLSA;
  - MultiARTM;
  - TransARTM.
- 2 Решение задачи классификации документов с использованием распределения тем  $p(t | d)$  в качестве признаков.
- 3 Оценка качества восстановления матрицы  $\Theta$  путем вычисления точности решения задачи классификации документов:

$$\text{Точность} = \frac{1}{N} \sum_{i=1}^N [y_i^{\text{pred}} = y_i^{\text{true}}]$$



## Результаты

Число тем совпадает с заданным при генерации

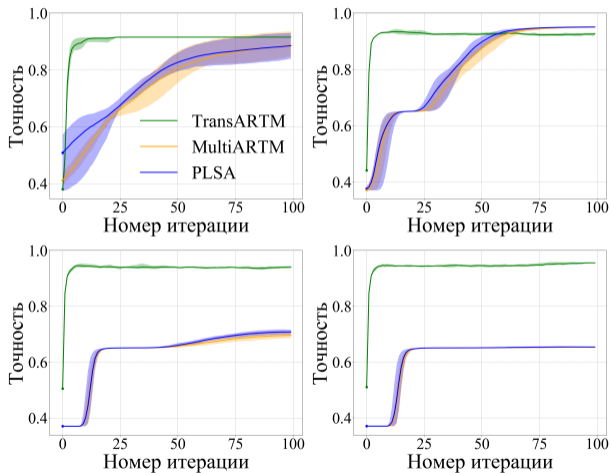


### Вывод:

TransARTM достигает высокого качества быстрее других моделей на транзакционных данных.

## Результаты

Меняем число тем  
от 5 до 100

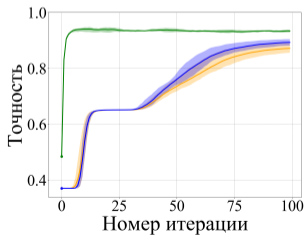
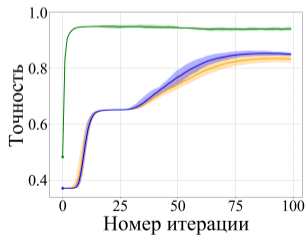
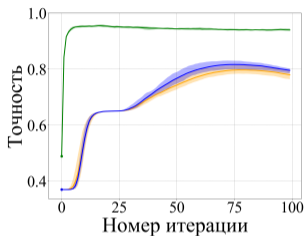
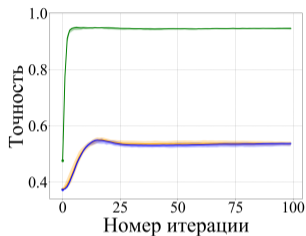


### Вывод:

TransARTM наиболее устойчива относительно инициализации и выбора числа тем.

## Результаты

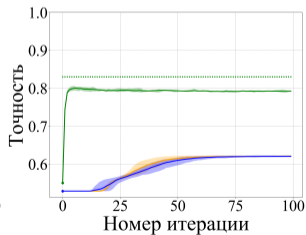
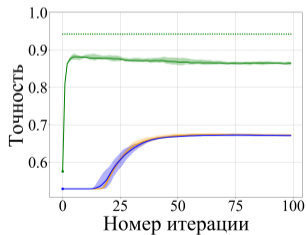
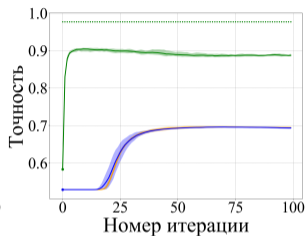
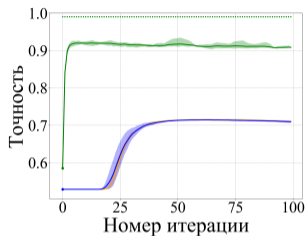
Меняем число  
транзакций  
от 450 000 до 13 500 000



### Вывод:

TransARTM

восстанавливает  
изначальную структуру  
матрицы  $\Theta$  с высоким  
качеством даже при  
небольшом числе данных.

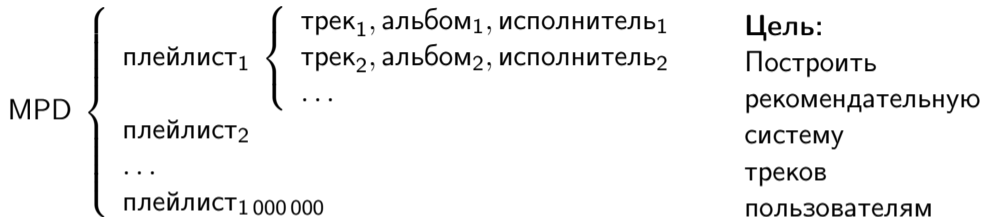


## Результаты

Меняем разреженность  
матрицы  $\Theta$   
от 0.2 до 0.8

**Вывод:**  
TransARTM показывает  
близкое к максимальному  
качество вне зависимости  
от разреженности  
матрицы  $\Theta$ .

# The Million Playlist Dataset (MPD)



	MPD	Train	Test	Valid
Число плейлистов:	1 000 000	100 000	1 000	1 000
Число треков:	66 346 428	9 875 306	232 613	232 808
Число уникальных треков:	2 262 292	296 882	39 368	38 641
Число уникальных альбомов:	734 684	140 983	20 690	20 483
Число уникальных артистов:	295 860	69 280	10 081	10 008

## Цели эксперимента:

- 1 Применить TransARTM к задаче построения рекомендаций.
- 2 Проанализировать различные гипотезы порождения плейлиста.
- 3 Сравнить результаты с другими моделями.

## Постановка эксперимента:

- 1 Для нахождения параметров моделей используем обучающую выборку, состоящую из 100 000 плейлистов, каждый из которых содержит от 140 до 250 треков.
- 2 Настраиваем коэффициенты регуляризации по сетке на валидационной выборке.
- 3 Предсказываем ранжированный список из 500 треков для каждого плейлиста тестовой выборки (последние 70 каждого плейлиста используются для оценки качества).
- 4 Используем следующие метрики для оценки качества:
  - **precision**;
  - **recall**;
  - **fscore**;
  - **ndcg** — normalized discounted cumulative gain.

Тем: 500

Модель	Рассматриваемые взаимодействия	Метрики, @500			
		precision	recall	fscore	ndcg
TopTracks	-	0.0230	0.1646	0.0404	0.1152
PLSA	(Pl, Tr)	0.0592	0.4228	0.1038	0.3025
LDA	(Pl, Tr)	0.0583	0.4162	0.1022	0.2988
MultiARTM	(Pl, Al), (Pl, Tr)	0.0594	0.4245	0.1043	0.3029
	<b>(Pl, Ar), (Pl, Tr)</b>	<b>0.0608</b>	<b>0.4343</b>	<b>0.1067</b>	<b>0.3110</b>
	(Pl, Ar), (Pl, Al), (Pl, Tr)	0.0605	0.4321	0.1061	0.3098
TransARTM	(Pl, Al, Tr)	0.0490	0.3497	0.0859	0.2484
	<b>(Pl, Ar, Tr)</b>	<b>0.0504</b>	<b>0.3603</b>	<b>0.0885</b>	<b>0.2555</b>
	(Pl, Al, Tr), (Pl, Ar, Tr)	0.0502	0.3587	0.0879	0.2548
	(Pl, Ar, Al, Tr)	0.0476	0.3398	0.0835	0.2374

**Вывод:**  
 TransARTM  
 показывает  
 сравнимые  
 результаты.

# Результаты

- 1 Предложено обобщение методов тематического моделирования на случай, когда исходные данные представимы в виде гиперграфа.
- 2 Проведены эксперименты на модельных данных, демонстрирующие корректность предложенного метода и преимущество его использования для сложноструктурированных данных.
- 3 Продемонстрировано применение гиперграфовой многомодальной тематической модели для построения рекомендательной системы.

## Доклады:

- 1 Международная научная конференция «Ломоносов-2018», «Многомодальные тематические модели на гиперграфах».
- 2 Data Fest<sup>5</sup>, «Гиперграфовые тематические модели для анализа транзакционных данных».



TransARTM

Гиперграф  
Постановка  
задачи  
Алгоритм

Эксперимент

Модельные  
данные  
Постановка  
Результаты  
Реальные  
данные  
Постановка  
Результаты

Заключение



## Fast online EM-algorithm for TransARTM

**ProcessBatch** iterates  $d \in D_b$  at a constant  $\Phi$ .

**Input:** set of vertices-containers  $D_b$ , matrix  $\Phi$ ;

**Output:** matrix  $(\tilde{n}_{vtk})$ ;

- 1:  $\tilde{n}_{vtk} := 0$  for all  $v \in V, t \in T, k \in K$ ;
- 2: **for all**  $d \in D_b$  **do**
- 3:     initialize  $\theta_{td} := \frac{1}{|T|}$  for all  $t \in T$ ;
- 4:     **repeat**
- 5:          $p_{tdx} = \mathop{\text{norm}}_{t \in T} \left( \theta_{td} \prod_{v \in x} \varphi_{vtk} \right)$  for all  $x \in d_k, t \in T, k \in K$ ;
- 6:          $n_{td} = \sum_{k \in K} \sum_{x \in d_k} \tau_k n_{dx} p_{tdx}$  for all  $t \in T$ ;
- 7:          $\theta_{td} = \mathop{\text{norm}}_{t \in T} \left( n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}} \right)$  for all  $t \in T$ ;
- 8:     **until**  $\theta_{td}$  converges;
- 9:      $\tilde{n}_{vtk} := \tilde{n}_{vtk} + \sum_{x \in d_k} [v \in x] \tau_k n_{dx} p_{tdx}$  for all  $v \in V, t \in T$   
 $k \in K$ ;

Probabilistic process of data generation using considered model:

---

**Input:**  $K$ , distributions  $p(t | d)$ ,  $p_k(v | t)$ , for all  $k \in K$ ;

**Output:** edges of the hypergraph (transactions);

- 1: **for all**  $k \in K$  **do**
  - 2:     define  $D_k$ ;
    - ▷  $K$  is the set of edge types
    - ▷  $D_k$  is the set of vertices-containers
  - 3:     **for all**  $d \in D_k$  **do**
    - 4:         define the number of hyperedges —  $n_d$ ;
    - 5:         **for all**  $i = 1, \dots, n_d$  **do**
      - 6:              $d_i := d$ ;
      - 7:             choose random topic  $t_i$  from  $p(t | d_i)$ ;
      - 8:             **for all**  $j = 2, \dots, h(k)$  **do**
        - 9:                 choose random object  $v_i$  from  $p_k(v | t_i)$ ,  $v \in V_{m_{kj}}$ ;
          - ▷  $h(k) = |e|$ ,  $e \in E_k$

# The Million Playlist Dataset (MPD)

The dataset contains 1 000 000 playlists:

- pid** – playlist id;
- name** – the name of the playlist;
- description** – the description given to the playlist;
- modified\_at** – timestamp when this playlist was last updated;
- num\_artists** – the total number of unique artists;
- num\_albums** – the number of unique albums;
- num\_tracks** – the number of tracks in the playlist;
- num\_followers** – the number of followers this playlist;
- num\_edits** – the number of separate editing sessions;
- duration\_ms** – the total duration of all the tracks in the playlist;
- collaborative** – if true, the playlist is a collaborative playlist;
- tracks** – an array of information about each track in the playlist:
  - track\_name** – the name of the track;
  - track\_uri** – the Spotify URI of the track;
  - album\_name** – the name of the track's album;
  - album\_uri** – the Spotify URI of the album;
  - artist\_name** – the name of the track's primary artist;
  - artist\_uri** – the Spotify URI of track's primary artist;
  - duration\_ms** – the duration of the track in milliseconds;
  - pos** – the position of the track in the playlist (zero-based).

Each vertex  $v \in V$  has **modality**  $m = \mu(v) \in M$ :

$$V = \bigsqcup_{m \in M} V_m, \text{ where } V_m = \{v \in V : \mu(v) = m\}.$$

Each edge  $e \in E$  has the **transaction type**  $k = \varkappa(e) \in K$ :

$$E = \bigsqcup_{k \in K} E_k, \text{ where } E_k = \{e \in E : \varkappa(e) = k\}.$$

All edges of type  $k$  have the same:

- ① degree  $h = h(k) = |e|$ , where  $e = \{v_1, \dots, v_h\}$ ;
- ② set of vertices' modalities:  $\mu(v_j) = m_{kj}, j = 1, \dots, h$ .

Edge's type  $k$  corresponds to the discrete probability space:

$$\Omega_k = V_{m_{k1}} \times \dots \times V_{m_{kh}} \times T$$

with a probability function  $p_k: \Omega_k \rightarrow [0, 1]$ .

Each vertex  $v$  is associated with latent topics:

$$p_k(v, t) = p_k(v | t)p_k(t) = p_k(t | v)p_k(v).$$

Probability distribution normalized within each modality:

$$\sum_{v \in V_m} p_k(v) = 1; \quad \sum_{v \in V_m} p_k(v | t) = 1.$$

- 1 For each type of edges  $k$  first modality  $m_{k1}$  is a container.

$D_k$  is the set of all vertices-containers of type  $k$  edges.

$d_k$  is the set of edges  $x: (d, x) \in E_k$ , where  $d \in D_k$ .

- 2 The distribution of the topics in vertex-container  $d$  are not depends on the edge type:

$$p_k(t | d) = p(t | d) \text{ for all } d \in D_k.$$

- 3 The hypothesis of conditional independence of vertices:

$$p_k(x | t, d) = p_k(x | t) = \prod_{v \in x} p_k(v | t).$$



## Hypergraphical Topic Model

The mathematical model is the following:

$$\begin{aligned}
 p_k(d, x) &= p_k(x | d) p_k(d) = p_k(d) \sum_{t \in T} p_k(x | t) p_k(t | d) = \\
 &= p_k(d) \sum_{t \in T} p(t | d) \prod_{v \in X} p_k(v | t) = p_k(d) \sum_{t \in T} \theta_{td} \prod_{v \in X} \varphi_{vtk}.
 \end{aligned}$$

Model's parameters:

- 1 Conditional probability of vertices in topics:  $\varphi_{vtk} = p_k(v | t)$ ;
- 2 Conditional probability of topics in the containers  $\theta_{td} = p(t | d)$ ;
- 3 The probability  $p_k(d)$  estimated from observed data:

$$p_k(d) = \frac{\sum_{x \in d_k} n_{dx}}{\sum_{d' \in D_k} \sum_{x \in d'_k} n_{d'x}}.$$

# Hypergraphic Topic Model (TransARTM)

The considered hypergraphic topic model is defined by:

- 1 The oriented hypergraph  $G = \langle V, E \rangle$ ;
- 2 The set of modalities  $M$ ;
- 3 The decomposition of the vertices set into subsets of different modalities  $\mu: V \rightarrow M$ ;
- 4 The set of edge types  $K$ ;
- 5 The decomposition of the edges set into subsets of different edges type  $\varkappa: E \rightarrow K$ ;
- 6 Degree of  $h(k)$  and set of modalities  $m_{k1}, \dots, m_{kh(k)}$  of type  $k$  edges;
- 7 The set of topics  $T$ ;
- 8 The probability space  $\Omega_k$  with the distribution  $p_k$  for all  $k \in K$ ;
- 9 The model parameters  $\varphi_{vtk} = p_k(v | t)$  and  $\theta_{td} = p(t | d)$ .

## Theoretical justification

**Theorem 1.** If the function  $R(\Phi, \Theta)$  is continuously differentiable and  $(\Phi, \Theta)$  is the local maximum of the considered problem, then the system of equations for the model's parameters  $\varphi_{vtk}$ ,  $\theta_{td}$  and auxiliary variables  $p_{tdx}$ ,  $n_{td}$  and  $n_{vtk}$  holds.

Let's use Karush–Kuhn–Tucker conditions and write the Lagrangian of the optimization problem:

$$\begin{aligned} \mathcal{L}(\Phi, \Theta) = & \sum_{k \in K} \tau_k \sum_{d \in D_k} \sum_{x \in d_k} n_{dx} \ln \sum_{t \in T} \theta_{td} \prod_{v \in X} \varphi_{vtk} + R(\Phi, \Theta) - \\ & - \sum_{k \in K} \sum_{m \in M} \sum_{t \in T} \lambda_{kmt} \left( \sum_{v \in V_m} \varphi_{vtk} - 1 \right) - \sum_{k \in K} \sum_{m \in M} \sum_{v \in V_m} \sum_{t \in T} \lambda_{kmvt} \varphi_{vtk} - \\ & - \sum_{k \in K} \sum_{d \in D_k} \mu_d \left( \sum_{t \in T} \theta_{td} - 1 \right) - \sum_{k \in K} \sum_{d \in D_k} \sum_{t \in T} \mu_{td} \theta_{td}. \end{aligned}$$

## Theoretical justification

Equate to zero the derivatives of a Lagrangian for the model parameters:

$$\frac{\partial \mathcal{L}}{\partial \varphi_{vtk}} = \sum_{d \in D_k} \sum_{x \in d_k} [v \in x] \tau_k n_{dx} \frac{\theta_{td} \prod_{u \in x \setminus v} \varphi_{utk}}{p_k(d, x)} + \frac{\partial R}{\partial \varphi_{vtk}} - \lambda_{k\mu(v)t} - \lambda_{k\mu(v)vt} = 0;$$

$$\frac{\partial \mathcal{L}}{\partial \theta_{td}} = \sum_{k \in K} \sum_{x \in d_k} \tau_k n_{dx} \frac{\prod_{v \in x} \varphi_{vtk}}{p_k(d, x)} + \frac{\partial R}{\partial \theta_{td}} - \mu_d - \mu_{td} = 0.$$

Multiply left and right sides of the first equality by  $\varphi_{vtk}$ , the left and right side of the second equality by  $\theta_{td}$ :

$$\sum_{d \in D_k} \sum_{x \in d_k} [v \in x] \tau_k n_{dx} \underbrace{\frac{\theta_{td} \prod_{u \in x} \varphi_{utk}}{p_k(d, x)}}_{p_{tdx} = p(t | d, x)} + \varphi_{vtk} \frac{\partial R}{\partial \varphi_{vtk}} = \lambda_{k\mu(v)t} \varphi_{vtk};$$

$$\sum_{k \in K} \sum_{x \in d_k} \tau_k n_{dx} \underbrace{\frac{\theta_{td} \prod_{v \in x} \varphi_{vtk}}{p_k(d, x)}}_{p_{tdx} = p(t | d, x)} + \theta_{td} \frac{\partial R}{\partial \theta_{td}} = \mu_d \theta_{td}.$$

## Theoretical justification

Rewrite these equations in the variables  $n_{vtk}$  and  $n_{td}$ :

$$\varphi_{vtk} \lambda_{k\mu(v)t} = n_{vtk} + \varphi_{vtk} \frac{\partial R}{\partial \varphi_{vtk}}. \quad (1)$$

$$\theta_{td} \mu_d = n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}; \quad (2)$$

If we assume that  $\lambda_{kmt} \leq 0$ , then the regularity condition does not hold, and in this case, according to the agreement,  $\varphi_{vtk} = 0$  for each  $v \in V_m$ . If the dual variable  $\lambda_{kmt}$  is positive, then both side of the equation (1) is non-negative. Combining these two cases into one formula, we obtain:

$$\varphi_{vtk} \lambda_{k\mu(v)t} = \max\left\{n_{vtk} + \varphi_{vtk} \frac{\partial R}{\partial \varphi_{vtk}}, 0\right\}.$$

Analogically, if  $\mu_d \leq 0$ , then the regularity condition does not hold, and according to the agreement,  $\theta_{td} = 0$  for all  $t \in T$ . If  $\mu_d > 0$ , then both side of the equation (2) is non-negative. Combining these two cases into one formula, we obtain:

$$\theta_{td} \mu_d = \max\left\{n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}, 0\right\}.$$

## Theoretical justification

Sum the left and right side of the equation (3) by  $v \in V_m$ , the left and right side of the equation (4) by  $t \in T$ , and apply normalization conditions and express the dual variables:

$$\mu_d = \sum_{t \in T} \max \left\{ n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}, 0 \right\};$$

$$\lambda_{kmt} = \sum_{v \in V_m} \max \left\{ n_{vtk} + \varphi_{vtk} \frac{\partial R}{\partial \varphi_{vtk}}, 0 \right\}.$$

Substituting the obtained expressions of dual variables in (3) and (4), we get desired equations for E and M step. The theorem is proved. ■