Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное
учреждение высшего профессионального образования
«Московский физико-технический институт
(государственный университет)»

Факультет управления и прикладной математики

Кафедра «Интеллектуальные Системы»

Построение и оценка качества гетерогенных иерархических тематических моделей

Выпускная квалификационная работа
(бакалаврская работа)

Направление подготовки: 03.03.01 Прикладные математика и физика

Выполнил:
студент 474 группы ____________________ Селезнева Мария Сергеевна

Научный руководитель:
к.ф.-м.н. ____________________ Воронцов Константин Вячеславович

Москва 2018

Оглавление

Todo list

Глава 1

Введение

Вероятностное тематическое моделирование — это раздел машинного обучения, решающий задачу поиска тем в коллекции документов. Тематическая модель определяет, к каким темам относится каждый документ и какие слова образуют темы [1]. Базовыми подходами построения тематических моделей являются PLSA [2] и LDA [3]. В работах [4, 5, 6] предложены модификаций данных подходов, учитывающие специфику конкретных задач. Аддитивная регуляризация тематических моделей (ARTM)[7, 8, 9, 10, 11] позволяет комбинировать упомянутые модели, интерпретируя их как регуляризаторы в PLSA.

В больших текстовых коллекциях темы часто образуют иерархии, в которых каждая тема делится на более специфичные подтемы. Моделью такой ситуации являются тематические иерархии. Они удобны для навигации пользователей по коллекциям, поэтому являются подходящей моделью для агрегирования контента.

Общепринятого определения и подхода к построению иерархических тематических моделей не существует. В модели иерархического LDA (hLDA) [12] темы образуют дерево. С другой стороны, модель иерархического распределения патинко (hPAM)[13] и модель иерархического ARTM (hARTM) [14] представляют собой направленный ациклический многодольный граф, что лучше соответсвует реальным отношениям между темами в мультидисциплинарных научных и научно-популярных статьях.

Модель hARTM — это развитие идеи аддитивной регуляризации тематических моделей для задачи построения тематических иерархий. Она позволяет применять регуляризацию как к уровням иерархии для комбинирования любых тематических моделей, так и к самой иерархии для контроля разреженности отношения «родитель-ребенок».

Цель данной работы — построение иерархической тематической модели по нескольким источникам в рамках модели hARTM. Построение модели по объединенной коллекции источников, различных по объему и тематической структуре, не решает поставленной задачи, так как темы, уникальные для меньшего из источников, теряются. В работе предлагается дополнять существующую модель одного источника выборками документов из новых источников. Дополнение происходит в два этапа: сначала выбираются документы нового источника, наиболее подходящие для добавления в коллекцию, затем выбранные документы добавляются в коллекцию и строится дополненная модель. При этом в силу стратегии инциализации и регуляризации модели существующие темы сохраняются, а добавленные документы уточняют соответсвующие им темы на первом уровне иерархии. На втором уровне добавленные документы могут порождать подтемы, характерные только для нового источника.

В работах [15, 16] рассмотрено построение иерархических тематических моделей гетерогенных источников . Однако, задача последовательного достроения модели затронута в работах, не использующих подход вероятностного тематического моделирования [17]. Предложенный в данной работе подход выигрывает у построения по объединенной коллекции по качеству решения и скорости.

Для проведения экспериментов используется BigARTM — библиотека для тематического моделирования с открытым исходным кодом [18, 19].

Глава 2

Постановка задачи

В вероятностном тематическом моделировании коллекция документов рассматривается как множество троек (d, w, t), выбранных случайно и независимо из дискретного распределения $p(d, w, t)$, заданного на конечном множестве $D \times W \times T$. Документы $d \in D$ и термины $w \in W$ являются наблюдаемыми переменными, тема $t \in T$ является латентной (скрытой) переменной. Построить тематическую модель коллекции документов D — значит найти распределения $p(w|t)$ для всех тем $t \in T$ и распределения $p(t|d)$ для всех документов $d \in D$. Известными при это являются распределения $p(w|d)$ терминов (токенов) в документах коллекции.

Предполагается, что порядок токенов $w \in W$ в документе $d \in D$ не важен (гипотеза "мешка слов"), что позволяет представить коллекцию в виде матрицы $[n_{dw}]_{D \times W}$, где
n_{dw} — число вхождений w в d. Также коллекцию можно представить в виде матрицы эмпирических оценок вероятности встретить токен в документе $[p_{dw}]_{D \times W}$, где $p_{dw} = \hat{p}(w|d) = \frac{n_{dw}}{n_d}$, $n_d = \sum_{w \in W} n_{dw}$ — число слов в документе d. Кроме того, используется гипотеза условной независимости: $p(w|t, d) = p(w|t)$.

2.1 Плоская модель ARTM

При сделанных предположениях плоская (одноуровневая) тематическая модель описывается формулой

$$p(w|d) \approx \sum_{t \in T} p(w|t)p(t|d) \quad d \in D, w \in W, \qquad (*)$$

которая следует из определения условной вероятности и формулы полной вероятности. Пусть число тем $|T|$ много меньше числа документов $|D|$ и числа терминов $|W|$. Тогда представим задачу (*) в виде факторизации матрицы $F = [p(w|d)]_{W \times D}$:

$$F \approx \Phi\Theta.$$

Параметры модели — матрицы $\Phi = [\phi_{wt}]_{W \times T}$, $\phi_{wt} = p(w|t)$ (термины, относящиеся к темам) и $\Theta = [\theta_{td}]_{T \times D}$, $\theta_{td} = p(t|d)$ (темы документов) — такие, что

$$\sum_{w \in W} \phi_{wt} = 1, \ \sum_{t \in T} \theta_{td} = 1.$$

В ARTM матрицы Φ и Θ находятся с помощью максимизации суммы логарифма правдоподобия и регуляризаторов R_i с неотрицательными коэффициентами регуляризации τ_i [10]:

$$\sum_{d \in D} \sum_{w \in W} n_{dw} \ln \sum_{t \in T} \phi_{wt}\theta_{td} + \sum_i \tau_i R_i(\Phi, \Theta) \to \max_{\Phi, \Theta}$$

$$\text{при условиях } \phi_{wt} \geq 0, \ \theta_{td} \geq 0, \ \sum_{w \in W} \phi_{wt} = 1, \ \sum_{t \in T} \theta_{td} = 1.$$

2.2 Мультимодальная модель ARTM

Документы $d \in D$ могут содержать не только текст, но и другие элементы, такие как тэги, ссылки, имена авторов и т.д. Такие типы элементов называются модальностями. Пусть M — множество модальностей. Каждой $m \in M$ соответсвует отдельный словарь (множество токенов) W^m, причем $W = \bigsqcup_{m \in M} W^m$.

Пусть $F^m = [p(w|d)]_{W^m \times D}$, $m \in M$ — матрицы наблюдаемых вероятностей для каждой модальности, а Φ^m — соответсвующие матрицы скрытых вероятностей $p(w|t)$. Определим F и Φ как объединения строк F^m и Φ^m, $m \in M$ соответственно. Тогда получим задачу факторизации $F \approx \Phi\Theta$ для мультимодальной тематической модели.

При построении мультимодальной тематической модели максимизируется взвешенная сумма логарифмов правдоподобия для всех модальностей $m \in M$ и регуляризаторов R_i [10]:

$$\sum_{m \in M} \kappa_m \sum_{d \in D} \sum_{w \in W^m} n_{dw} \ln \sum_{t \in T} \phi_{wt}\theta_{td} + \sum_i \tau_i R_i(\Phi, \Theta) \to \max_{\Phi, \Theta}$$

$$\text{при условиях } \phi_{wt} \geq 0, \ \theta_{td} \geq 0, \ \sum_{w \in W} \phi_{wt} = 1, \ \sum_{t \in T} \theta_{td} = 1.$$

Как показано в [7], эта оптимизационная задача решается EM-алгоритмом.

E-шаг:

$$p(t|d,w) = p_{tdw} = \operatorname{norm}_{t\in T}(\phi_{wt}\theta_{td}).$$

M-шаг:

$$\phi_{wt} = \operatorname{norm}_{w\in W}\Big(n_{wt} + \phi_{wt}\sum_i \tau_i \frac{\partial R_i}{\partial \phi_{wt}}\Big); \quad n_{wt} = \sum_{d\in D} n_{dw}p_{tdw};$$

$$\theta_{td} = \operatorname{norm}_{t\in T}\Big(n_{td} + \theta_{td}\sum_i \tau_i \frac{\partial R_i}{\partial \theta_{td}}\Big); \quad n_{td} = \sum_{w\in d} n_{dw}p_{tdw}.$$

2.3 Регуляризаторы

Сглаживание и разреживание:

Регуляризатор сглаживания вводит в модель требование, чтобы столбцы ϕ_t и θ_d были близки к заданным распределениям $\beta_t = [\beta_{wt}]_{w\in W}$ и $\alpha_d = [\alpha_{td}]_{t\in T}$ в смысле дивиргенции Кульбака-Лейблера[10]:

$$R(\Phi,\Theta) = \beta_0 \sum_{m\in M}\sum_{t\in T}\sum_{w\in W^m} \beta_{wt}\ln\phi_{wt} + \alpha_0 \sum_{d\in D}\sum_{t\in T}\alpha_{td}\ln\theta_{td} \to \max_{\Phi,\Theta},$$

где β_0 и α_0 — заданные положительные коэффициенты.

Регуляризатор разреживания имеет такой же вид, но коэффициенты β_0 и α_0 отрицательны. Он способствует обращению значительной части вероятностей ϕ_{wt} и θ_{td} в ноль, что соответствует естественному предположению о том, что каждый документ d и каждый токен w связаны лишь с небольшим числом тем t.

Декоррелирование:

Регуляризатор декоррелирования минимизирует ковариации между столбцами ϕ_t, что способствует увеличению различности тем модели [10]:

$$R(\Phi) = -\frac{\tau}{2}\sum_{t\in T}\sum_{s\in T\backslash t}\operatorname{cov}(\phi_t,\phi_s) \to \max_{\Phi,\Theta},\ \operatorname{cov}(\phi_t,\phi_s) = \sum_{w\in W}\phi_{wt}\phi_{ws}.$$

2.4 Иерархическая модель ARTM

Тематическая иерархия ARTM представляет собой многоуровневый граф, где каждый уровень — это плоская тематическая модель. Для ее построения необходимо не только строить тематические модели уровней, но и устанавливать связи родитель-ребенок между темами соседних уровней. Для этого вводятся специальные межуровневые регуляризаторы.

Пусть построено $l \geq 1$ уровней тематической иерархии, параметры l-того уровня — матрицы Φ^l, Θ^l, A — множество тем l-го уровня. Построим $(l+1)$-ый уровень с параметрами Φ, Θ и множеством тем T.

Будем моделировать распределение токенов по темам l-того уровня как смесь распределений по темам $(l+1)$-го уровня [14]:

$$p(w|a) \approx \sum_{t \in T} p(w|t)p(t|a) \quad a \in A, w \in W.$$

Это приводит к задаче факторизации

$$\Phi^l \approx \Phi\Psi, \tag{**}$$

где $\Phi = [p(w|t)]_{W \times T}, \Psi = [p(t|a)]_{T \times A}$. Полученная матрица Ψ содержит распределения тем $(l+1)$-го уровня в темах l-го уровня. Таким образом, межуровневый регуляризатор — это логарифм правдоподобия для задачи (**):

$$R(\Phi, \Psi) = \sum_{a \in A} \sum_{w \in W} n_{wa} \ln \sum_{t \in T} \phi_{wt}\psi_{ta} \to \max_{\Phi, \Psi},$$

где $n_a = \sum\limits_{w \in W} n_{wa}$ — веса тем родительского уровня, пропорциональные их размеру.

Этот регуляризатор эквивалентен добавлению в коллекцию $|A|$ псевдодокументов, представленных матрицей $[n_{wa}]_{W \times A}$. Тогда Ψ образует $|A|$ дополнительных столбцов матрицы Θ, соответсвующих этим псевдодокументам.

Глава 3

Proposed Framework

3.1 Notation and Problem Setup

We consider the system of n thermostatically controlled loads, indexed by $i = 1, 2, \ldots, n$. We suppose that there is an aggregator A which send matrices $\bar{P}(t)$, $t = 1, \ldots, T$.

We refer $P_i(t) \in \mathbb{R}^{N\times N}$ as a transition matrix of load i, $i = 1, \ldots, n$, and N is a number of possible states of any load (all loads have the same number of states). Each load has its own behaviour defined as a function $P_i(t)$. All these behaviours are independent. We also suppose that any $\bar{P}(t)$ and $P_i(t) \in \mathcal{P} = \{P^{(1)}, P^{(2)}, \ldots, P^{(m)}\}$. Each load is allowed to accept or reject a transition matrix $\bar{P}(t)$ according to the following rule:

$$\bar{P}_i(t) = \begin{cases} \bar{P}(t), & \text{if } \Omega(\bar{P}(t), P_i(t), t) = 1 \\ P_i(t), & \text{otherwise,} \end{cases}$$

where $\Omega(P_1, P_2, t)$ is some decision rule which meets all legal and customer requirements. For the sake of simplicity we may assume this rule is known (i.e. a part of the contract), however, it is not required: we may learn it on-the-fly. Also, as we have only m different matrices, Ω is a time-dependent matrix where $\Omega_{ij}(t) = 1$ if having own matrix $P_k(t) = P^{(i)}$ a load k accepts proposed matrix $\bar{P}^{(j)}$ at time t.

We refer $\pi_i(t)$ as a unit vector corresponding to the state of load i, $1 \le i \le n$, at time t, so that

$$\pi(t) = \frac{1}{n}\sum_{i=1}^{n} \pi_i(t).$$

We also assume the the power consumption q at each state is known, so that the total power consumption $s(t)$ is

$$s(t) = n \cdot \pi(t)^\top q.$$

We refer $\bar{s}(t)$ as the power requested by the system operator at time t.

We also assume the loss function of the system operator to be

$$\sum_{i=1}^{T} c(t)|\bar{s}(t) - s(t)|,$$

where $c(t) > 0$ is non-negative cost function.

In this research we implement an incentive-based model of users motivation to participate in the curtailment program. In particular we consider the Emergency Demand-Response Program (EDRP) setting here, in which consumers get incentive payments for reducing their power consumption during reliability triggered events (see [?]). Consumers may choose not to curtail and therefore to forgo the payments, which are usually specified beforehand [?]. Due to privacy protection reasons ([?]) the aggregator is forbidden to observe exact loads accepting a curtailment request, but it is important to know the total amount of these loads for two reasons:

1. As the aggregator's budget is strictly limited, it must estimate expenses for incentive payments.

2. Most of the DR incentive-based programs limit the total amount of curtailment hours to avoid user disturbance (typically 200 hours/year [?])

To sum up, the exact objective will be:

$$\min \sum_{i=1}^{T} c(t)|\bar{s}(t) - s(t)| \tag{3.1}$$

$$s.t. \sum_{t=1}^{T} k(t) \leq K \tag{3.2}$$

3.2 Solution

In the following section we treat 3.1 as a linear contextual bandit with knapsack problem [?]. Define the policy as some mapping from the context $f(t)$ to the matrix number $i \in \{0, 1, \ldots, m\}$ (a.k.a. arm). Define the regret $\mathcal{R}_T^{\nu}$ as the difference between performances under the best policy available and under some policy ν:

$$\mathcal{R}_T^\nu = \sum_{t=1}^{T}(r^*(t) - r^\nu(t)) \tag{3.3}$$

The goal is to learn the policy ν which minimizes the regret:

$$\begin{aligned} &\min_\nu \mathcal{R}_T^\nu = \min_\nu \sum_{t=1}^{T}(r^\nu(t) - r^*(t)) \\ &s.t. \sum_{t=1}^{T} k(t) \leq K \end{aligned} \tag{3.4}$$

where the penalty is defined as

$$r^\nu(t) = c(t)|s^\nu(t) - \bar{s}(t)|$$

and $k(t)$ is the number of loads who accepted the curtailment request at the time t, K is the contract limit for user disturbance and ν is the policy being used. Hereafter we refer to $r^\nu(t)$ as $r(t)$.

In this form of bandits setup we face two major troubles:

1. General bandit with knapsack problem setup [?] consider penalties as a random independent variables over time and arms. Both assumptions here are false: our choices from the past strongly affects the current reward distributions.

2. Variables $k(t)$ are unobservable, hence we can only ensure thresholds with some probability.

To deal with these issues consider the the mechanics of how $s(t)$ depends on aggregator's actions. Suppose that by the time t we have a (possibly unknown) state-distribution $\pi(t)$. It is easy to see that if each device chooses its own matrix, and we denote the number of devices which have chosen matrix P_j as n_j, then the next-moment consumption with no control will be:

$$s(t|0) := \sum_{j=1}^{m} n_j(t) q^T P_j \pi(t) = \sum_{j=1}^{m} n_j(t) f_j(t|0) \tag{3.5}$$

and if we pull i^{th} arm, the consumption will be:

$$s(t|i) := \sum_{j=1}^{m} n_j(t) q^T (\Omega_{ij}(t) P_i + (1 - \Omega_{ij}(t)) P_j) \pi(t) = \sum_{j=1}^{m} n_j(t) f_j(t|i) \tag{3.6}$$

The model appears to be linear over unknown variables $n_j(t)$ and some vector $f(t|i) := \{f_j(t|i)\}_{j=1}^m$, $i \in [0, 1, \ldots m]$ which we know if we have $\pi(t)$ and $\Omega_{ij}(t)$. Therefore it is surprisingly convenient to consider $f(t|i)$ as a feature vector of an arm i at the moment t. Note, that the features $f_j(i|t)$ have the natural interpretation as the amount of energy being consumed by an average device which has $\bar{P} = P_j$ at the time t if we pull the i-th arm. Moreover, we also have an estimator for budgeted variable: $k(t|i) = \sum_{i=1}^m \Omega_{ij} n_j$.

The Bandit The above mentioned arm feature vectors (a.k.a. arm-contexts) $f(i|t)$ reveals a straightforward "linear contextual bandit with knapsack"setup:

$$\begin{aligned} &\min_{\nu: t \to \{0,1\ldots m\}} \sum_{t=1}^T \mathcal{L}_t\left(n(t)^T f(t|\nu(t))\right) \\ &s.t. \, \mathbb{E} \sum_{t=1}^T k(t|\nu(t)) \le K \\ &\sum_{j=1}^m n_j(t) = n \, \forall t \\ &n_j(t) \ge 0 \, \forall t \end{aligned}$$

where $n(t) = \{n_j(t)\}_{j=1}^m$ (do not miss it with n – the total amount of devices), and $\mathcal{L}_t(x)$ is a loss function. In this work two classical demand-response loss functions are considered. The first one is a trivial mapping:

$$\mathcal{L}_t(x) = x \tag{3.7}$$

which gives the consumption minimization setup:

$$\begin{aligned} &\min_{\nu: t \to \{0,1\ldots m\}} \sum_{t=1}^T n(t)^T f(t|\nu(t)) = \min_{\nu} \sum_{t=1}^T s^\nu(t) \\ &s.t. \, \mathbb{E} \sum_{t=1}^T k(t|\nu(t)) \le K \\ &\sum_{j=1}^m n_j(t) = n \, \forall t \\ &n_j(t) \ge 0 \, \forall t \end{aligned} \tag{3.8}$$

and the second one is the absolute deviation from a given series $\bar{s}(t)$:

$$\mathcal{L}_t(x) = |x - \bar{s}(t)| \tag{3.9}$$

which gives the consumption stabilization setup:

$$\begin{aligned} \min_{\nu: t \to \{0,1\ldots m\}} \sum_{t=1}^{T} |n(t)^T f(t|\nu(t)) - \bar{s}(t)| &= \min_{\nu} \sum_{t=1}^{T} |s^{\nu}(t) - \bar{s}(t)| \\ s.t.\, \mathbb{E} \sum_{t=1}^{T} k(t|\nu(t)) &\leq K \\ \sum_{j=1}^{m} n_j(t) &= n \,\forall t \\ n_j(t) &\geq 0 \,\forall t \end{aligned} \tag{3.10}$$

It remains to notice that $n(t)$ is a periodic function over time with a period of 24 hours (at least within one season of a year), so we can consider 3.8 and 3.10 as a set of 24 bandits: each one is learning to make a decision at the assigned hour. Note that these problems are in exact form for recently emerged budgeted bandit solvers ([?] and more (add later)), and all the requirements are met: the divergence between predicted and real consumption may appear only regarding to stochatical fluctuations due to the finiteness of the ensemble, hence we have independence of reward as a function of a context through arms and time (of course we need a proof here).

The Algorithm with known $\Omega_{ij}(t)$ and without a knapsack: Consider a simpler setup when $K = \infty$ i.e. we have no knapsack constraints in our problem. Here τ is the number of steps which a device performs each hour. We also assume that we do not know neither an initial state distribution $\pi(0)$ nor $n_j(t)$. In this case 1 is how a dummy algorithm may look like (we assume here a 3.9 loss function).

One may notice that in 1 the greedy policy was used, and no exploration steps were proposed. In fact, the greedy policy here turns to be optimal: all the reward variances depend only on unknown vector $n(t)$ and when we learn it we decrease reward variances for all arms simultaneously. Hence, in contrast to the following case, no exploration-exploitation balance is required here.

The Algorithm without a knapsack and without $\Omega_{ij}(t)$ Note that it is not crucial for the previous algorithm to know the oracle $\Omega_{ij}(t)$, as we

Algorithm 1 Non-budgeted TCL-control with oracle

Require: $\mathcal{P}$, n, $\tau, \Omega_{ij}(t)$

$n_i(0) := 1/n \, \forall i$

$\pi_0 := \frac{1}{n}\sum_{i=1}^{m} n_i(0)u_i$, where u_i is a stationary distribution of the matrix P_i

$A(t) = \{\emptyset\} \, \forall t$ where A(t) is an array of all feature vectors through time which we got at the hour t. It will be used as a learning dataset for the t-th regressor.

$S(t) = \{\emptyset\} \, \forall t$ where S(t) is an array of all s(t) through time which we got at the hour t. These are the targets for the t-th regressor.

for $t := 1 \ldots T$ **do**

  Generate the arm contexts $\{f_j(t|i)\}_{j=1}^{m}$ for all $i \in \{0, 1 \ldots m\}$:

  $f_j(t|i) := \sum_{\xi=1}^{\tau} q^T(\Omega_{ij}(t)P_i^{\xi} + (1-\Omega_{ij}(t))P_j^{\xi})\pi(t) \quad i \in \{1 \ldots m\}$

  $f_j(t|0) := \sum_{\xi=1}^{\tau} q^T P_j^{\tau}\pi(t)$

  if $\bar{s}(t) == 0$ **then**

    Set $i := 0$ (no control) and send it to the ensemble

  else

    $i := \arg\max_{i\in\{1\ldots m\}} \mathcal{L}_t(s(t|i)) := \arg\max_{i\in\{1\ldots m\}} \mathcal{L}_t(n(t)^T f(t|i))$.

    Send the chosen arm i to the ensemble

  end if

  Wait and receive $s(t|i)$ – the actual ensemble consumption

  $A(t)$.append$(f(t|i))$

  $S(t)$.append$(s(t|i))$

  Solve a constrained regression problem to learn real $n_j(t)$:

  $n(t) = \arg\min_{n\in R_+^n} \|A(t)n(t) - S(t)\|^2$, $s.t. \sum_{j=1}^{m} n_j(t) = n$, $n_j(t) \geq 0$, start from $n(t)$

  Calculate next π:

  $\pi(t+1) = \frac{1}{n}\sum_{j=1}^{m} n_j(t)(\Omega_{ij}(t)P_i^{\tau} + (1-\Omega_{ij}(t))P_j^{\tau}))\pi(t)$

  if $n_j(t+1)$ are undefined yet **then**

    $n_j(t+1) = n_j(t)$

  end if

end for

may learn in the same way we learn $n(t)$. Formally, let i be the arm the aggregator pulled at the moment t, then the consumption will be:

$$\begin{aligned} s(t|i) =& n(t)^T f(t|i) = \sum_{j=1}^{m} n_j(t) \sum_{\xi=1}^{\tau} q^T (\Omega_{ij}(t) P_i^{\xi} + (1 - \Omega_{ij}(t)) P_j^{\xi}) \pi(t) = \\ &= \sum_{j=1}^{m} \Omega_{ij}(t) n_j \sum_{\xi=1}^{\tau} q^T (P_i^{\xi} - P_j^{\xi}) \pi(t) + \underbrace{\sum_{j=1}^{m} n_j \sum_{\xi=1}^{\tau} q^T P_j^{\xi} \pi(t)}_{s(t|0)} \end{aligned} \tag{3.11}$$

and we have a discrete regularized regression problem over the vector $\Omega_i(t) \in \{0, 1\}^m$ here:

$$\underbrace{s(t|i) - s(t|0)}_{target} = \sum_{j=1}^{m} \underbrace{\Omega_{ij}(t)}_{variables} \underbrace{n_j(t) \sum_{\xi=1}^{\tau} q^T (P_i^{\xi} - P_j^{\xi}) \pi(t)}_{features} \tag{3.12}$$

If we also have observations from i-th arm at t-th hour from the past (i.e. this is not the first time we pull this arm at this hour), then we may solve the least-squared optimization problem over $\Omega_{ij}(t)|_{j=j} \in \{0, 1\}^m$ which might reduce the fluctuations of Ω.

The algorithm 2 for this case is the essentially the same: the new parts are marked with red.

The Algorithm with a knapsack and without $\Omega_{ij}(t)$ This one is our final result. Note that we just need to apply not a regular UCB1 bandit inside the abovementioned algorithm, but the "bandit-with-knapsack"solver from [?] or something similar.

Algorithm 2 Non-budgeted TCL-control without oracle

Require: $\mathcal{P}$, n, τ, $\cancel{\Omega_{ij}(t)}$

$n_i(0) := 1/n \, \forall i$
$\pi_0 := \frac{1}{n}\sum_{i=1}^{m} n_i(0) u_i$
$A(t) = \{\emptyset\} \, \forall t$
$S(t) = \{\emptyset\} \, \forall t$
$\Omega_{ij}(0) = \{0,1\}^{m \times m}$ – random matrix
for $t := 1 \dots T$ **do**
  Generate arm contexts $\{f_j(t|i)\}_{j=1}^{m}$ for all $i \in \{0, 1 \dots m\}$:
  $f_j(t|i) := \sum_{\xi=1}^{\tau} q^T (\Omega_{ij}(t) P_i^{\xi} + (1 - \Omega_{ij}(t)) P_j^{\xi}) \pi(t) \quad i \in \{1 \dots m\}$
  $f_j(t|0) := \sum_{\xi=1}^{\tau} q^T P_j^{\tau} \pi(t)$
  if $\bar{s}(t) == 0$ **then**
    Set $i := 0$ (no control) and send it to the ensemble
    Wait and receive $s(t|0)$ – the actual ensemble consumption
    When we have no control signal, we learn $n_j(t)$
    $A(t)$.append($f(t|0)$)
    $S(t)$.append($s(t|0)$)
    Solve a constrained regression problem to learn real $n_j(t)$:
    $n(t) = \arg\min_{n \in R_+^n} \|A(t)n(t) - S(t)\|^2$, $s.t. \sum_{j=1}^{m} n_j(t) = n$, $n_j(t) \geq 0$, start from $n(t)$
    Calculate next π:
    $\pi(t+1) = \sum_{j=1}^{m} n_j(t) P_j^{\tau} \pi(t)$
  else
    Calculate $s(t|i) = n(t)^T f(t|i)$ for all $i \in \{0, 1, \dots m\}$
    Calculate variance estimations $\Delta s(t|i)$
    $i := \arg\min_{j \in \{0,1,\dots m\}} \mathcal{L}_t(s(t|j) + B\sqrt{\Delta s(t|i)})$
    Send the chosen arm i to the ensemble
    Wait and receive $s(t|i)$ – the actual ensemble consumption
    Learn the vector $\Omega_i(t)$ solving optimization problem:
    $\Omega_i(t) := \arg\min_{\omega \in \{0,1\}^m} \Bigg(\underbrace{(s(t|i) - s(t|0))}_{s} - \sum_{j=1}^{m} \omega_j \, n_j(t) \underbrace{\sum_{\xi=1}^{\tau} q^T (P_i^{\xi} - P_j^{\xi}) \pi}_{g_j}$
    $= \arg\min_{\omega \in \{0,1\}^m} \left(s - w^T g\right)^2$
    I suppose we need some kind of smooth relaxation here
    Calculate next π:
    $\pi(t+1) = \frac{1}{n}\sum_{j=1}^{m} n_j(t) (\Omega_{ij}(t) P_i^{\tau} + (1 - \Omega_{ij}(t)) P_j^{\tau})) \pi(t)$
  end if
  if $n(t+1)$ is undefined yet **then**
    $n(t+1) = n(t)$
  end if
  if $\Omega_{ij}(t+1)$ is undefined yet **then**
    $\Omega_{ij}(t+1) = \Omega_{ij}(t)$
  end if

Глава 4

Numerical Experiments

Глава 5

Conclusion

Литература

[1] David M. Blei, “Probabilistic topic models”, Commun. ACM, vol. 55, no. 4, pp. 77–84, 2012.

[2] T. Hoffman, “Probabilistic latent semantic indexing”, in Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 50–57. ACM Press, New York, 1999.

[3] David M. Blei, Andrew Y. Ng, and Michael I. Jordan, “Latent dirichlet allocation”, Journal of Machine Learning Research, vol. 3, pp. 993–1022, 2003.

[4] Michal Rosen-Zvi, Thomas L. Griffiths, Mark Steyvers, and Padhraic Smyth, “The author-topic model for authors and documents”, CoRR, vol. abs/1207.4169, 2012.

[5] Chaitanya Chemudugunta, Padhraic Smyth, and Mark Steyvers, “Modeling general and specific aspects of documents with a probabilistic topic model”, in NIPS, Bernhard Schölkopf, John C. Platt, and Thomas Hofmann, Eds. 2006, pp. 241–248, MIT Press.

[6] Khoat Than and Tu Bao Ho, “Fully sparse topic models”, in ECML/PKDD (1), Peter A. Flach, Tijl De Bie, and Nello Cristianini, Eds. 2012, vol. 7523 of Lecture Notes in Computer Science, pp. 490–505, Springer.

[7] Konstantin Vorontsov, Oleksandr Frei, Murat Apishev, Peter Romov, Marina Suvorova, and Anastasia Yanina, “Non-bayesian additive regularization for multimodal topic modeling of large collections”, in TM@CIKM, Nikolaos Aletras, Jey Han Lau, Timothy Baldwin, and Mark Stevenson, Eds. 2015, pp. 29–37, ACM.

[8] Konstantin Vorontsov, Anna Potapenko, and Alexander Plavin, “Additive regularization of topic models for topic selection and

sparse factorization", Statistical Learning and Data Sciences, January 2015.

[9] K. V. Vorontsov, "Additive regularization for topic models of text collections", Doklady Mathematics, vol. 89, no. 3, pp. 301, May 2014.

[10] Konstantin Vorontsov and Anna Potapenko, "Additive regularization of topic models", Machine Learning, vol. 101, no. 1-3, pp. 303, October 2015.

[11] Konstantin Vorontsov and Anna Potapenko, "Tutorial on probabilistic topic modeling: Additive regularization for stochastic matrix factorization", in Analysis of Images, Social Networks and Texts. Springer, January 2014.

[12] D. M Blei, T. Griffiths, Michael I. Jordan, and J. Tenenbaum, "Hierarchical topic models and the nested chinese restaurant process", 2003.

[13] David M. Mimno, Wei Li 0010, and Andrew McCallum, "Mixtures of hierarchical topics with pachinko allocation", in ICML, Zoubin Ghahramani, Ed. 2007, vol. 227 of ACM International Conference Proceeding Series, pp. 633–640, ACM.

[14] NA Chirkova and KV Vorontsov, "Additive regularization for hierarchical multimodal topic modeling", 2016.

[15] Chi Wang, Jialu Liu, Nihit Desai, Marina Danilevsky, and Jiawei Han, "Constructing topical hierarchies in heterogeneous information networks", Knowl. Inf. Syst, vol. 44, no. 3, pp. 529–558, 2015.

[16] Marina Danilevsky, Chi Wang, Fangbo Tao, Son Nguyen, Gong Chen, Nihit Desai, Lidan Wang, and Jiawei Han, "AMETHYST: A system for mining and exploring topical hierarchies of heterogeneous data", December 03 2013.

[17] Nachiketa Sahoo, Jamie Callan, Ramayya Krishnan, George T. Duncan, and Rema Padman, "Incremental hierarchical clustering of text documents", in CIKM, Philip S. Yu, Vassilis J. Tsotras, Edward A. Fox, and Bing Liu 0001, Eds. 2006, pp. 357–366, ACM.

[18] Konstantin Vorontsov, Oleksandr Frei, Murat Apishev, Peter Romov, and Marina Dudarenko, "Bigartm: Open source library

for regularized multimodal topic modeling of large collections", in Analysis of Images, Social Networks and Texts. Springer, January 2015.

[19] Oleksandr Frei and Murat Apishev, "Parallel non-blocking deterministic algorithm for online topic modeling", in Analysis of Images, Social Networks and Texts. Springer, January 2017.