Министерство науки и высшего образования Российской Федерации
МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ
(государственный университет)

Факультет управления и прикладной математики

Базовая организация — ФИЦ ИУ РАН,
Вычислительный центр им. А.А. Дородницына РАН

Кафедра «Интеллектуальные системы»
специализация «математические и информационные технологии (ПиОС)»

Квалификационная работа на соискание степени бакалавра

по направлению 03.03.01 «Прикладные математика и физика»,

профиль «Компьютерные технологии и интеллектуальный анализ данных»

Тематическое моделирование в задаче классификации отзывов покупателей о работе и ассортименте продуктового магазина

Студент группы 473б — Павловская Анастасия Сергеевна

Научный консультант, д.ф-м.н. — Воронцов Константин Вячеславович

Научный руководитель, д.т.н. — Цурков Владимир Иванович

Москва, 2018

Содержание

Аннотация

Рассматривается задача классификации текстовых отзывов покупателей о работе продуктового магазина. В рамках решения поставленной задачи использовались метод опорных векторов, логистическая регрессия, градиентный бустинг на решающих деревьях и классификатор на основе тематической модели. Полученные результаты позволяют судить об эффективности применения тематических моделей в качестве генератора дополнительных признаков.

1 Введение

Задача классификации текстовых документов находит много приложений в современном мире. Её решают при фильтрации спама, при организации новостных потоков, при выделении сентиментов, в процессе рубрикации текстов в онлайн-бибилиотеках и в других областях. Задача классификации возникает и при оптимизации работы контакт-центров компаний.

Ежедневно компании получают тысячи сообщений от своих клиентов с просьбами, жалобами и предложениями по разным темам. Определение темы сообщения вручную требует колоссальных человеческих ресурсов. Целью данной работы является разработка интеллектуальной системы, призванной для каждого обращения определить его тему и адресовать его правильному специалисту. В настоящее время такие системы уже существуют, однако они основаны на простейших моделях машинного обучения, а следовательно, не дают высокого качества.

Для создания качественной системы классификации необходимо учитывать предметную область. В рамках данной работы рассматривается задача классификации отзывов покупателей продовольственного магазина, темы обращений включают в себя жалобы о качестве продуктов, предложения по улучшению функционирования торговых точек, благодарности

отдельным работникам магазинов.

В работе рассмотрены самые распространённые методы машинного обучения для решения задач классификации текстов: метод опорных векторов, логистическая регрессия и градиентный бустинг на решающих деревьях. Для каждого классификатора подбираются оптимальное признаковое описание данных и гиперпараметры. Новизна определяется использованием в качестве дополнительных признаков выделенных с помощью тематической модели тем.

Исследован и менее стандартный подход к задаче: классификация при помощи тематической модели. Тематическое моделирование – это способ построения модели коллекции текстовых документов, которая определяет, к каким темам относится каждый из документов и какие слова или словосочетания образуют каждую тему. В работе использовалась аддитивная регуляризация тематической модели (APTM)[1], которая позволяет учесть знания предметной области. Проведено исследование качества работы классификатора, выявлены плюсы и минусы в рамках конкретной задачи.

2 Теоретическое описание

2.1 Математическая постановка задачи классификации

Рассмотрим X – множество объектов, Y – множество ответов. Даны обучающая выборка $X^n = \{x_1, x_2, ...x_n\}$ и $\{y_1, y_2, ...y_n\}$ – множество ответов на этой выборке.

Пусть $f : X \to D_f$ задаёт некоторое отображение из пространства объектов в пространство допустимых значений признаков. Тогда вектор $x = (f_1(x), f_2(x), ..., f_k(x))$ называется признаковым описанием объекта $x \in X$. Матрица объекты-признаки имеет вид:

$$F = \|f_j(x_i)\|_{ji} = \begin{pmatrix} f_1(x_1) & ... & f_k(x_1) \\ ... & ... & ... \\ f_1(x_n) & ... & f_k(x_n) \end{pmatrix}$$

Требуется найти алгоритм a: $\forall i\ a(f(x_i)) = \mathbf{p}_i = \left(p_{i_0}... \ p_{i_{|Y|}}\right)$ - вероятностное распределение объекта на множестве классов. Тогда объект будем относить к наиболее вероятному классу: $a_i = \arg\max_j (\mathbf{p}_{ij})$

Далее будем считать, что объекты задаются своими признаковыми описаниями, и матрицу F обозначать X, а i-ый объект выборки будем обозначать $\mathbf{x}_i = (x_{i1}, ..., \ x_{ik})^\top = (f_1(x_i), ..., \ f_k(x_i))^\top$.

В нашей задаче множество объектов – это коллекция текстовых документов D – отзывов покупателей о работе магазина – и дополнительная метаинформация. $Y = \{0, ... \ 16\}$ – множество меток классов.

Для оценки качества работы модели введём *функционцию потерь* $L(a, x)$ – неотрицательную функцию, которая характеризует величину ошибки алгоритма a на объекте x. *Функционал качества(средняя ошибка)* алгоритма

a на выборке X^n: $\frac{1}{n}\sum_{1}^{n} L(a, x_i)$

Введём дополнительные обозначения:

Таблица 1: My caption

		Верный ответ	
		1	0
Предсказание	1	TP (Верное решение)	FP (Ошибка второго рода)
	0	FN (Ошибка первого рода)	TN (Верное решение)

Для оценки качества работы классификаторов в данной работе использовались следующие функционалы:

1. accuracy:
$$\frac{1}{n}\sum_{i}[a_i = y_i]$$
где $a_i = \arg\max_j(\mathbf{p}_{ij})$, y_i – метка класса объекта x_i, а квадратные скобки обозначают индикатор: [истина] $= 1$, [ложь] $= 0$. Этот функционал показывает, какая доля объектов была правильно классифицирована. Качество модели тем лучше, чем больше значение функционала.

2. logloss:
$$-\frac{1}{n}\sum_{i}\sum_{j}[a_i = y_j]\log(p_{ij})$$
Чем меньше значение функционала, тем лучше модель предсказывает вероятность класса.

Последние два функционала определим для случая бинарной классификации и будем использовать для оценки качества классификации каждого класса.

3. precision:
$$\frac{TP}{TP + FP}$$
Для каждого класса функционал показывает, как часто классификатор делает верное предсказание, присваивая объекту метку данного класса. Функционал используется для оценки качества бинарной классификации, с его помощью будем сравнивать качество выделения разных классов.

4. recall:
$$\frac{TP}{TP + FN}$$
Для каждого класса функционал показывает, какая доля объектов этого класса была распознана с помощью модели. Функционал используется для оценки качества бинарной классификации, с его помощью будем сравнивать качество выделения разных классов.

2.2 Методы

Рассмотрим методы, которые были использованы для решения задачи в рамках данной работы.

2.2.1 Линейные модели

Опишем методы бинарной классификации и обобщим их на случай нескольких классов.

Пусть дана обучающая выборка $X^n = \{\mathbf{x}_i\}_{i=1}^n$ и множество ответов $\{y_i\}_{i=1}^n$, $y_i \in Y = \{-1, 1\}$. Необходимо найти вектор весов $\mathbf{w} \in \mathbb{R}^n$ – вектор параметров

линейной модели $a(\mathbf{x}, \mathbf{w}) = \text{sign}\left(\langle \mathbf{w}, \mathbf{x} \rangle - w_0\right)$. Геометрический смысл линейного классификатора состоит в том, что он строит разделяющую плоскость $\langle \mathbf{w}, \mathbf{x} \rangle - w_0 = 0$ и относит объекты по разные стороны от неё к разным классам.

Введём обозначение: $M_i = \left(\langle \mathbf{w}, \mathbf{x} \rangle - w_0\right) y_i$ – отступ объекта x_i. Если классификатор допустил ошибку на объекте, то $M_i < 0$, иначе $M_i > 0$. Тогда пользуясь данной терминологией, средняя ошибка на обучающей выборке:

$$\sum_i [a(\mathbf{x}_i) \neq y_i] = \sum_i [M_i < 0] \to \min_{\mathbf{w}}$$

Эмпирический риск является кусочно-постоянной функцией, которую сложно минимизировать, поэтому оценим его сверху непрерывной функцией и будем оптимизировать полученную оценку.[2]

Метод опорных векторов[3] Будем использовать следующую аппроксимацию средней ошибки:

$$\sum_i [a(x_i) \neq y_i] = \sum_i [M_i < 0] \leq \sum_i (1 - M_i)_+ \leq \sum_i (1 - M_i)_+ + \frac{1}{2C}\|\mathbf{w}\|^2 \qquad (*)$$

Здесь запись $(1 - M_i)_+$ обозначает положительную срезку($|x|_+ = max\{0, x\}$). Алгоритм, оптимизирующий полученный функционал называется методом опорных векторов. Смысл слагаемого $\frac{1}{2C}\|\mathbf{w}\|^2$ будет пояснён ниже.

Поясним геометрический смысл полученного метода. Для этого введём понятие *оптимальной* разделяющей плоскости – плоскость, расстояние до которой от объектов обоих классов максимально. Задача построения оптимальной разделяющей плоскости эквивалентна задаче оптимизации функционала $(*)$.

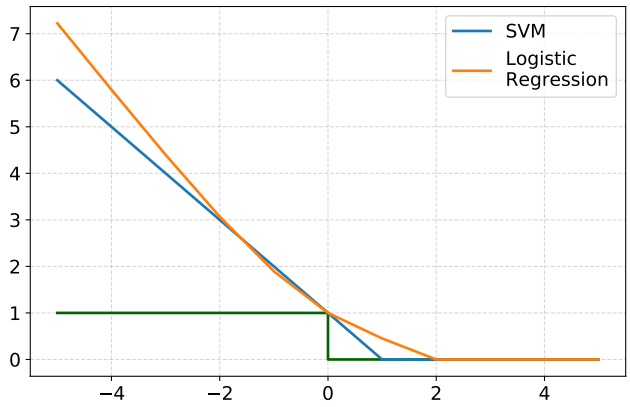

Рис. 1: Аппроксимации пороговой функции потерь

Логистическая регрессия Воспользуемся другой аппроксимацией средней ошибки:

$$\sum_i [M_i < 0] \leq \sum_i \log\left(1 + \exp\{M_i\}\right)$$

Данная модель имеет интересное свойство: минимизация аппроксимации средней ошибки эквивалентна максимизации правдоподобия

$$P(y|x_i, w) = \frac{1}{1 + \exp\{-M_i\}}, \;\; M_i = y\left(\langle \mathbf{w}, \mathbf{x}_i \rangle - w_0\right)$$

Таким образом, обученная модель может предсказывать не только метки классов, но и апостериорные вероятности принадлежности объекта классу.

Регуляризация Будем называть задачу корректно поставленной по Адамару, если её решение существует, единственно и устойчиво. Задача поиска вектора параметров в линейной модели является некорректно поставленной по Адамару задачей. Для решения такого рода часто используется регуляризация – добавление дополнительной информации или дополнительных ограничений на оптимизируемый функционал.

Используемые в работе виды регуляризаторов:

1. $L2$-регуляризатор[4]:

$$\|\mathbf{w}\|_2^2 = \sum_i w_i^2$$

С помощью данного ограничения можно бороться с переобучением модели. Переобучение – явление, при котором модель строит верные предсказания для обучающей выборки, однако делает много ошибок классификации на тестовом наборе данных. Проявляется это явление часто в том, что веса модели имеют большие по модулю значения. Для борьбы с переобучением в классификатор добавляется регуляризатор, то есть накладывается штраф на слишком большие веса.

2. $L1$-регуляризатор:

$$\|\mathbf{w}\|_1 = \sum_i |w_i|$$

. Данный регуляризатор имеет способность отбирать признаки, обнуляя веса перед ними.[5]

2.2.2 Градиентный бустинг над решающими деревьями

Решающее дерево Эта модель представляет из себя дерево, в каждой вершине которого записано условие, а в листе – метка класса. В данной работе рассматривается бинарное решающее дерево.

Для построения дерева используется жадный способ: в каждой вершине происходит разбиение выборки на две части в соответствие с условием $[\mathbf{x}_{ij} \leq t]$. Объекты, удовлетворяющие этому условию образуют выборку X^l и идут в одно из поддеревьев, а не удовлетворяющие ему X^r – в другое. При этом разбиение останавливается, если в листе дерева оказалось меньше, чем $min\_data\_in\_leaf$ объектов.[6]

Параметры условия для разбиения подбираются при оптимизации функ-

ционала ошибки:

$$Q\left(X^n, j, t\right) \to \min_{j,t}$$

В качестве функционала ошибки в данной работе выбран критерий вида:

$$Q\left(X^n, j, t\right) = \frac{|X^l|}{|X^n|} H(X^l) + \frac{|X^r|}{|X^n|} H(X^r)$$

Здесь X^r, X^l – подвыборки в правой и левой дочерних вершинах соответственно, $H(X)$ – критерий информативности (такая функция, значение которой тем меньше, чем меньше разброс ответов в X). В качестве критерия информативности в данной работе используется критерий Джини:

$$H(X) = \sum_{k=1}^{|Y|} q_k(1 - q_k), \; q_k = \frac{1}{n} \sum_{i \in X} [y_i = k]$$

Градиентный бустинг [7] Данный метод заключается в построении композиции алгоритмов таким образом, что каждый следующий алгоритм компенсирует ошибки уже существующего ансамбля. Таким образом, финальный алгоритм имеет вид: $a(x_i) = \sum\limits_t g_t(x_i)$, где g_t – базовый алгоритм. Пусть на первом шаге построения композиции задан базовый алгоритм a_0, обученный на выборке $\{x_i, y_i\}_{i=1}^n$; пусть также на шаге k задана композиция алгоритмов $a(x_i) = \sum\limits_t^k g_t(x_i)$. Тогда для построения следующего элемента композиции обучим базовый алгоритм на выборке $\{-L'(y_i, \sum\limits_t^k g_t(x_i)), x_i\}_{i=1}^n$, где $L(a, x)$ – дифференцируемая функция ошибки.

2.2.3 Обобщение на случай многоклассовой классификации

В данной работе будем использовать метод One-vs-all. Для этого построим $|Y|$ бинарных классификаторов, каждый из которых будет работать на выборке вида $X = (\mathbf{x}_i)_{i=1}^n, \{[y_i = k]\}_{i=1}^n \forall k \in Y$. Каждый построенный классификатор строится на тех же объектах, однако метки классов бинаризуются.

Таким образом, каждый из классификаторов строит разделяющую плоскость $\langle \mathbf{w}_k, \mathbf{x} \rangle - w_{k_0} = 0$. Метка класса для объекта x_i определяется по следующему правилу: $a_i = \arg\max_{k \in Y} (\langle \mathbf{w}_k, \mathbf{x}_i \rangle - w_{k_0})$

2.2.4 Тематическая модель [8]

Пусть D - коллекция документов, W - словарь всех слов в документах, T - конечное множество тем. Каждый отдельный документ $d \in D$ представляет собой набор слов $w_1, w_2, ... \in W$. Каждое вхождение терма w связано с некоторой темой $t \in T$.

Воспользуемся гипотезой мешка слов, которая предполагает, что порядок слов в документе можно не учитывать и что вероятность слова зависит от темы и не зависит от документа. Используя формулу полной вероятности, получаем вероятность встретить слово в документе:

$$p(w \mid d) = \sum_{t \in T} p(w \mid t) p(t \mid d) = \sum_{t \in T} \varphi_{wt} \theta_{td}$$

В матричной записи:

$$F = \Phi\Theta$$

$F = (p(w|d))_{W \times D}$ - матрица частот слов в документах

$\Phi = (\varphi_{wt})_{w \in W, t \in T}$ - матрица термов тем

$\Theta = (\theta_{td})_{t \in T, d \in D}$ - матрица тем документ

Основная задача тематического моделирования $p(w \mid d)$ найти $\varphi_{wt}, \theta_{td}$. Или, в терминах матричного разложения, представить матрицу F в виде произведения двух стохастических матриц: $F \approx \Phi\Theta$.

Введём *функцию правдоподобия* – совместное распределение выборки как функцию от параметров модели:

$$p((d_i, w_i)_{i=1}^n; \Phi, \Theta) = \prod_{i=1}^{n} p(d_i, w_i) = \prod_{d \in D} \prod_{w \in d} p(w|d)^{n_{dw}} p(d)^{n_{dw}}$$

Исходя из принципа максимума правдоподобия будем выбирать такие параметры модели, при которых выборка наиболее правдоподобна. Прологорифмировав правдоподобие, переходим от произведения к сумме и отбрасываем слагаемые, не зависящие от параметров модели. Таким образом, получаем оптимизационную задачу:

$$\sum_{d\in D}\sum_{w\in d} n_{wd}\ln(\sum_{t\in T}\varphi_{wt}\theta_{td}) \to \max_{\Phi,\Theta}$$

$$\sum_{w\in W}\varphi_{wt} = 1;\ \varphi_{wt} \geq 0;\ \sum_{t\in T}\theta_{td} = 1;\ \theta_{td} \geq 0$$

n_{wd} - количество вхождений слова w в документ d

Регуляризация Для ограничения сложности модели к основному оптимизируемому функционалу добавляется ограничения согласно предметной области. Таким образом, оптимизируемый функционал принимает вид:

$$\sum_{d\in D}\sum_{w\in d} n_{wd}\ln(\sum_{t\in T}\varphi_{wt}\theta_{td}) + R(\Phi,\Theta) \to \max_{\Phi,\Theta};\ R(\Phi,\Theta) = \sum_i \tau_i R_i$$

$$\sum_{w\in W}\varphi_{wt} \in \{0,1\};\ \varphi_{wt} \geq 0;\ \sum_{t\in T}\theta_{td} \in \{0,1\};\ \theta_{td} \geq 0$$

τ_i - коэффициент регуляризации

R_i - регуляризатор

Рассмотрим регуляризаторы, использованные в данной работе.

1. Одним из основных преимуществ тематических моделей является интерпретируемость полученных в результате работы тем. Из требования интерпретируемости естественным образом получается требование разреженности матриц $\Phi,\ \Theta$. Другими словами, если мы хотим, чтобы тема была более понятна, то она должна описываться небольшим набором слов, а каждый документ должен содержать небольшой

набор тем. Для сглаживания и разреживания матриц Φ, Θ вводится регуляризатор:

$$R(\Phi, \Theta) = \sum_{t \in T} \sum_{w \in W} \beta_{wt} \ln \varphi_{wt} + \sum_{d \in D} \sum_{t \in T} \alpha_{td} \ln \theta_{td}$$

Положительное значение коэффициентов α_{td}, β_{wt} соответствует сглаживанию, а отрицательное значение – разреживанию.

2. Наложим ещё одно ограничение на модель: она должна быть информативна, а значит темы должны как можно меньше коррелировать между собой. Введём регуляризатор декоррелирования:

$$R\left(\Phi\right) = -\frac{\tau}{2} \sum_{t \in T} \sum_{s \in T \setminus t} \sum_{w \in W} \varphi_{wt} \varphi_{ws}$$

Мультимодальная модель Документы включают в себя не только текстовые описания, но и дополнительные метаданные, которые помогают определить тематику. Каждый тип таких данных образует отдельную модальность со своим словарём. Примерами могут служить указания авторов[9], жанров и категорий[10], [11], цитируемых документов[12], графических изображени и так далее.

Пусть M – множество модальностей, словари которых $\{W_m\}_i^{|M|}$ попарно не пересекаются. Тематическая модель каждой модальности будет иметь такой же вид, как и тематическая модель, введённая ранее: $p_{wd} = \sum_{t \in T} \varphi_{wt} \theta_{td},\ w \in W_m, m \in M, d \in D$. Таким образом, для каждой модальности можно построить матрицу $\Phi_m = \left(\varphi_{wt}\right)_{w \in W_m, t \in T}$. Записав полученные матрицы в столбец, получим матрицу Φ; распределение тем в документе оставим общим для всех модальностей.

При постановки задачи оптимизации будем брать взвешенную линейную комбинацию прологарифмированных функций правдоподобий каждой мо-

дальности:

$$\sum_{m\in M}\tau_m\sum_{d\in D}\sum_{w\in W_m}n_{dw}\ln\sum_{t\in T}\varphi_{wt}\theta_{td}+\sum_i\tau_i R_i\to\max_{\Phi,\Theta}$$

Классификация с помощью тематической модели Построим модель с модальностями текста W и класса C. Для обучения будем использовать подвыборку документов, для каждого из которых известен класс или подмножество классов $C_d \in C$. Затем будем подавать модели документы, для которых класс не известен и для предсказания использовать линейную вероятностную модель классификации: $p(c|d) = \sum_{t\in T}\varphi_{ct}\theta_{td}$. При этом документ d будем относить к классу c, если $p(c|d) \geq \gamma_c$. Коэффициенты φ_{ct} и γ_c настраиваются по выборке документов с известными классами, а признаковое описание документа θ_d вычисляется только по его термам.

3 Практическое исследование

3.1 Описание предметной области

Для полного понимания специфики задачи рассмотрим существующий процесс получения и обработки сообщений пользователей.

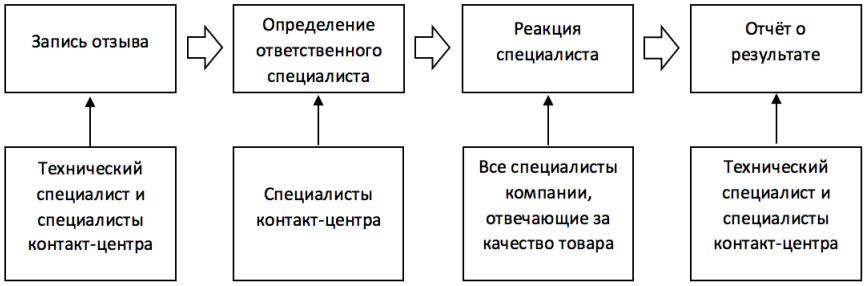

Рис. 2: Цепочка обработки сообщения

Таким образом, главная задача – по отзыву определить специалиста, который поможет покупателю справиться с возникшей трудностью, или исправит проблему.

3.2 Описание данных

Дана коллекция D ($|D| = 58267$) документов, каждый из которых представляет из себя отзыв покупателя о магазине, товарах. Коллекция размечена: каждый отзыв отнесён к единственной категории.

Список категорий: инородные предметы в продуктах, испортившиеся раньше срока годности товары, недолив, недовес; отрицательные изменения в качестве (консистенция изменилась, раньше был лучше, много соли или сахара, не нравится производитель); не нравится вкус продукта; проблемы с упаковкой (например, мятая упаковка); проблемы с работой магази-

на (некомпетентный продавец, очередь, нарушение санитарных норм, технические неудобства); благодарность за продукт; вопросы и предложения по составу продуктов; пожелания по этикетке и упаковке (очень перекликается с категорией "проблемы с упаковкой"); наличие товара; просьба вернуть/оставить продукт, низкая/высокая цена; жалобы и предложения по бонусам и акциям; просьбы об открытии торговой точки, планировка магазина; проблемы с оборудованием; жалобы и благодарности. Основная сложность классификации связана с тем, что категории пересекаются по своему смыслу, поэтому сложно произвести хорошую классификацию.

Коллекция имеет иерархию: каждая категория разбивается на подкатегории. Например, заявки по работе магазина касаются очереди, технческих неудобств и так далее. Так как ключевая задача – определить специалиста, до которого надо донести сведения о проблеме, то важно не допускать ошибок при определении классов верхнего уровня. Количество сообщений со временем растёт [Рис. 4], выборка является несбалансированной [Рис. 3].

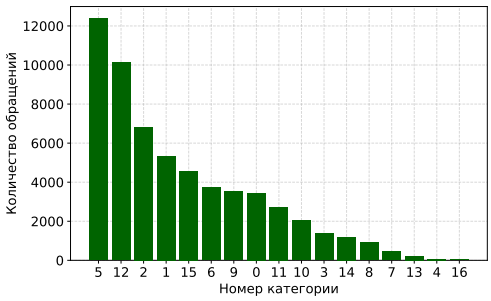

Рис. 3: Количество обращений в каждой категории

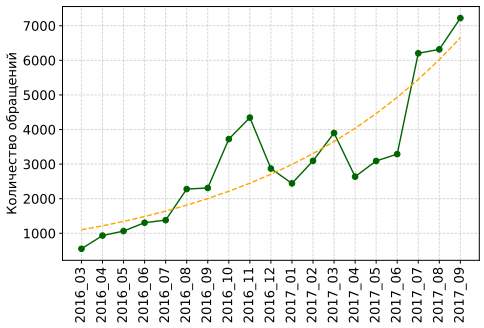

Рис. 4: Распределение количества обращений по времени

Некоторые категории имеют больший вес при классификации. Значимость класса определяется на основе специфики предметной области. На-

пример, принципиально важно не ошибаться при определении класса отзывов, связанных с качеством продуктов. Поэтому необходимо уменьшать ошибку первого рода в приоритетных категориях за счёт повышения ошибки второго рода в менее приоритетных (то есть при возникновении спорной ситуации объект лучше отнести к приоритетной категории).

Приоритетные категории: категория 0, категория 2, категория 5. В дальнейшем, качество работы моделей на этих категориях будет выделяться жирным шрифтом.

Признаки объектов Для каждого документа d коллекции D доступна следующая информация: дата обращения, источник, категория жалобы, текстовое описание, ответ производителя, продавец 1, продавец 2, продакт (специалист, отвечающий за введение нового продукта на рынок), производитель, результат, статус жалобы, торговая точка, технолог (специалист, отвечающий за качество поставляемого товара), тип включения, тип жалобы, товар.
В [Табл. 2] отображена статистика по пропущенным и уникальным значениям.

Не все признаки можно использовать для обучения модели. Например, признаки ответ производителя, результат и статус жалобы не известны на момент получения отзыва, а признаки продакт, технолог, товар и производитель сложно получить. Поэтому основной интерес для нас представляют признаки источник, торговая точка, продавец и текстовое описание.

Источники сообщений Сообщения поступают из разных каналов обратной связи [Рис.5]. Каждый источник обладает своей спецификой: отзывы на форумах длиннее и менее чётко сформулированы. Сообщения,

Признак	Процент пропусков	Уникальные значения
Продавец 2	**82%**	**509**
Тип включения	81%	2
Продавец 1	**79%**	**1670**
Ответ производителя	75%	4774
Результат	74%	17
Статус жалобы	71%	3
Продакт	63%	5
Технолог	57%	20
Торговая точка	**56%**	**741**
Производитель	52%	785
Товар	44%	2199
Источник	**0%**	**11**
Категория жалобы	0%	17
Описание	0%	39470
Тип жалобы	0%	72
Дата обращения	0%	37020

Таблица 2: Статистика по пропущенным и уникальными значениям

полученные из горячей линии имеют фиксированную структуру, так как записываются работниками контакт центра со слов покупателей.

Например:

> *Место выкладки: Теплые стеллажи нонфуд зафиксирована температура ниже +18. Температура на начало дня: 11 Температура в середине дня: 11,2; Температура на конец дня: 10,5*

> *Не списали купон от *** р по карте ***, начисление бонусами.*

Из [Рис. 6] можно сделать вывод, что хотя горячая линия и остаётся самым популярным способом связи, однако Telegramm — основной канал текстовых сообщений – также набирает популярность.

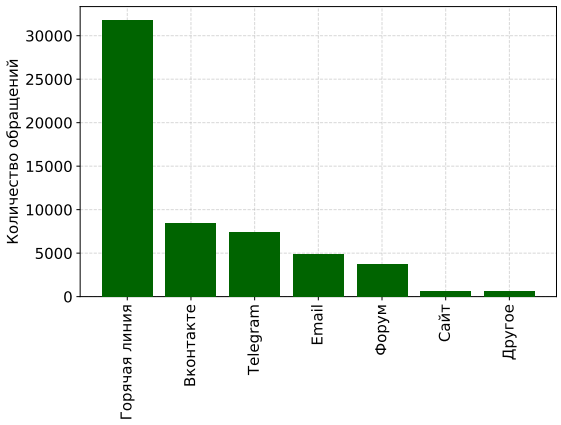

Рис. 5: Статистика по источникам обращений

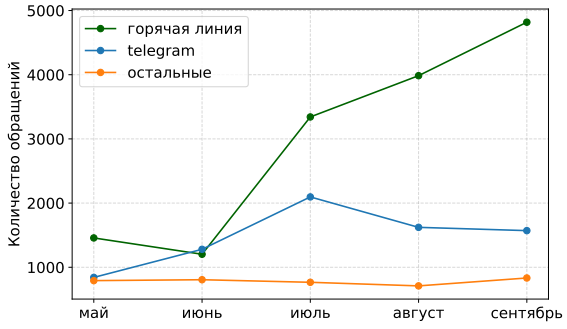

Рис. 6: Изменение популярности источников с течением времени

Текстовые описания Служат основным источником информации для модели. Средняя длина текста в символах – 204; средняя длина текста в словах – 33; общее число различных слов в документах – 90224. Из графика распределения длин документов в словах [Рис.7] можно сделать вывод, что в основном коллекция состоит из коротких текстов. Самые часто употребляемые слова: *температура*, *очень*, *магазин*, *день*, *это*, *сегодня*, *добрый*.

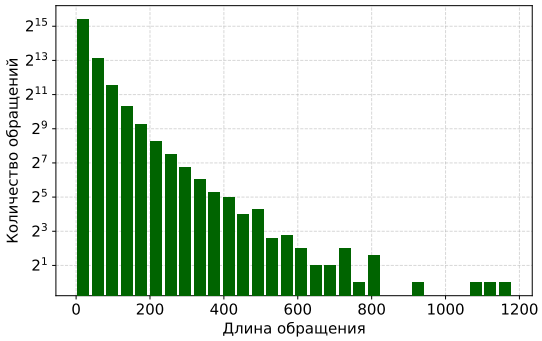

Рис. 7: Распределение длин документов в словах

Рис. 8: Средняя длина документов (в словах) в категории

Из графика распределения средних длин документов в категориях [Рис. 8] можно видеть, что 13 и 16 категории содержат исключительно короткие тексты. В основном это техническая информация.

3.3 Предобработка данных

Процесс предобработки данных состоит из следующих этапов:

1. *Удаление пустых записей.* Из данных удаляются записи с пустыми значениями в столбцах "Описание" , "Тип жалобы".

2. *Удаление записей операторов горячей линии.* Обращения, составленные операторами содержат специфичную лексику(например, *«дата изготовления *** в офис доставили ***»*), нет необходимости классифицировать их.

3. *Кодирование целевой переменной.* На данном этапе строится биекция между числами от 0 до 16 и названиями категорий.

4. *Лемматизация.* Приведение слов к нормальной (словарной) форме: для существительных – именительный падеж, единственное число; для прилагательных – именительный падеж, единственное число,

мужской род; для глаголов, причастий и деепричастий – глагол в инфинитиве.

5. *Добавление N-грамм.* Выделение из текстов часто встречающихся словосочетаний (например, *сумма бонусов на карте*, *номер карты*, *розничная цена*)

6. *Удаление стоп-слов.* Удаление часто встречающихся слов, которые не помогают в определении тематики документа(например, союзы, предлоги). К стандартному словарю шумовых слов были добавлены специфические для области стоп-слова.

7. *Фильтрация по длине документа.* Удаление документов, длина которых в словах меньше заранее определённой константы (в нашем случае она равна 3).

Признаковое описание объектов В исходном виде признаки имеют формат строк, поэтому не могут быть использованы в качестве признакового описания. Для представления объектов в понятной для алгоритма форме воспользуемся алгоритмами кодирования категориальных признаков и текстовых описаний.

Кодирование категориальных признаков выполняется с помощью алгоритма one-hot-кодирования. Категориальный признак, имеющий N значений, заменяется на N бинарных признаков: i-ый новый признак принимает значение 1 только на тех объектах, которые имели i-ое значение категориального признака. После one-hot-кодирования матрица объекты-признаки X_{categ} имеет размер $(58267, 10736)$. Для кодирования текстовых описаний использовалось два подхода:

1. *Модель «мешка слов»*. Документ $d_i \in D$ представляется вектором

x_i : $x_{ij} = n_j$ – число вхождений $w_j \in W$ (i-го слова из словаря $W : |W| = l$) в документ d_i

И матрица объекты-признаки имеет вид:

$$X_{bow} = \|x_{ij}\|_{n \times l}$$

2. *TF-IDF*. Документ $d_i \in D$ представляется вектором x_i:

$$x_{ij} = \text{tf-idf}(w_j, d_i, D) = \text{tf}(w_j, d_i) \times \text{idf}(w_j, D)$$

$$\text{tf}(w_j, d_i) = \frac{n_j}{\sum\limits_{k:w_k \in d_i} n_k}, \quad \text{idf}(w_j, D) = \log \frac{|D|}{|\{d \in D | w_j \in d\}|}$$

Где n_j – число вхождений w_j слова словаря в документ, $|D|$ – количество документов в коллекции, $|\{d \in D | w_j \in d\}|$ – количество документов в коллекции, в которых встречается w_j

И матрица объект-признак имеет вид:

$$X_{tfidf} = \|x_{ij}\|_{n \times |W|}$$

Таким образом, окончательная матрица объекты-признаки получается конкатенацией матрицы закодированных категориальных признаков X_{categ} и одной из матриц, кодирующих текстовое описание: X_{bow} или X_{tfidf}

3.4 Постановка эксперимента

Для выбора наиболее удачного признакового описания данных был проведён эксперимент: выборка разбивалась на 6 частей, по 5 из них модель обучалась, а на оставшейся - предсказывала. Далее разные описания сравнивались по распределению accuracy.

Для оценки качества классификации моделей и сравнения разных классификаторов выборка разбивается на обучающую и тестовую части. Разбиение производится в пропорции 5:1 таким образом, чтобы в пропорции

между классами сохранялись.

Модели сравниваются с помощью функционала *accuracy*. Кроме того, в каждой модели измеряется *precision*, *recall* классификации для каждого класса.

3.5 Используемые методы

3.5.1 Метод опорных векторов

Выбор признакового описания

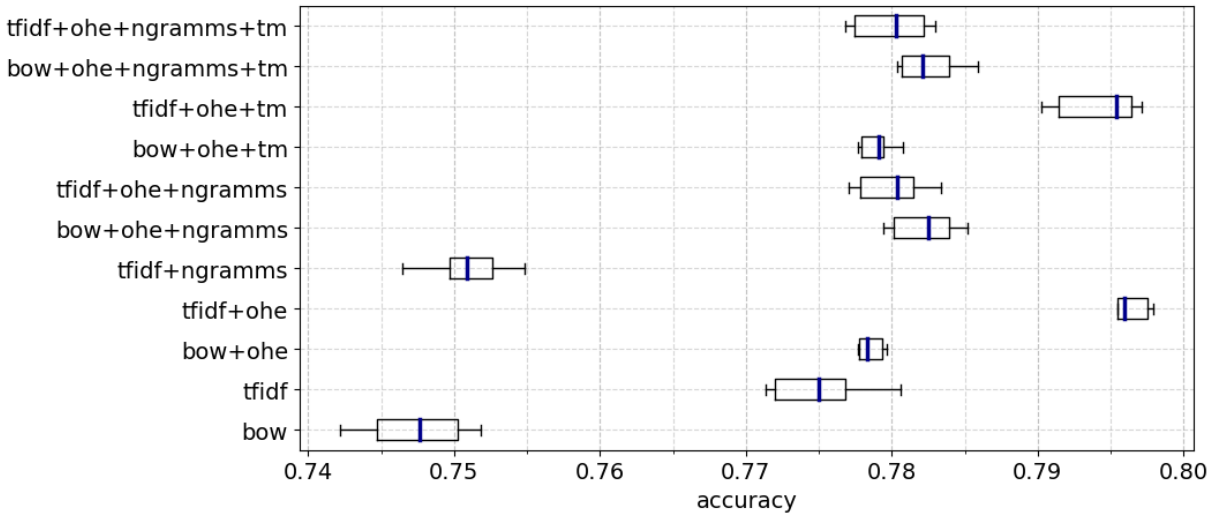

Рис. 9: Выбор оптимального признакового описания

Из графика можно сделать вывод, что оптимальным признаковым описанием будет tf-idf описание для текстов и бинарное кодирование для дополнительных признаков9.

Выбор параметров модели В данном пункте было построено две модели: с $l1$ и $l2$-регуляризациями соответственно. Затем, для каждой из мо-

делей производился подбор коэффициентов регуляризации по сетке:

$$[10^{-4}, 10^{-3}, 10^{-2}, 0.05, 0.1, 0.25, 0.75, 1.0, 2.5, 5.0, 7.5, 10.0, 100.0, 10^{3}, 10^{4}]$$

Оптимальная модель выбиралась с точки зрения максимизации accuracy. Оптимальная модель – это модель с l2-регуляризацией и константой $C = 0.75$.

Качество классификации оптимальной модели

accuracy = 0.8044

Категория	*5*	12	*2*	1	15	6	9	11	10	*0*
Precision	*0.93*	0.91	*0.66*	0.77	0.79	0.76	0.86	0.79	0.68	*0.87*
Recall	*0.94*	0.81	*0.75*	0.81	0.91	0.82	0.84	0.83	0.74	*0.89*

Категория	3	14	8	7	13	4	16
Precision	0.00	0.88	0.70	0.63	0.83	0.00	0.83
Recall	0.00	0.86	0.70	0.25	0.29	0.00	0.71

3.5.2 Логистическая регрессия

Выбор признакового описания

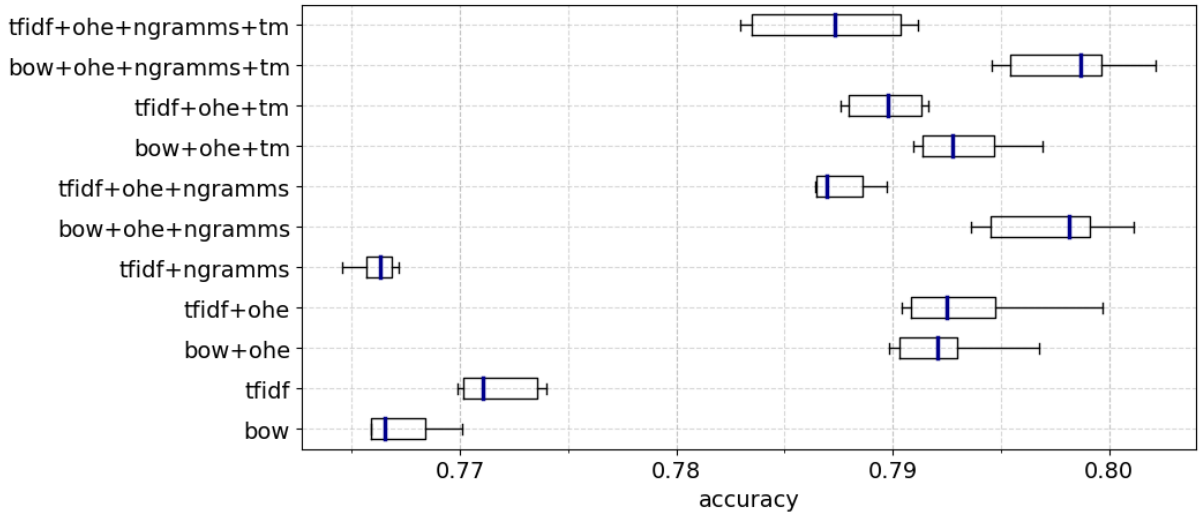

Рис. 10: Выбор оптимального признакового описания

Из графика можно сделать вывод, что оптимальным признаковым описанием будет «мешок слов» для текстов, утойчивые словосочетания, темы, выделенные тематической моделью, и бинарное кодирование для дополнительных признаков10.

Выбор параметров модели В данном пункте было построено две модели: с $l1$ и $l2$-регуляризациями соответственно. Затем, для каждой из моделей производился подбор коэффициентов регуляризации по сетке:

$$[10^{-4}, 10^{-3}, 10^{-2}, 0.05, 0.1, 0.25, 0.75, 1.0, 2.5, 5.0, 7.5, 10.0, 100.0, 10^{3}, 10^{4}]$$

Оптимальная модель – это модель с l2-регуляризацией и константой $C = 5.0$.

Качество классификации оптимальной модели

accuracy : 0.8014

Категория	5	12	2	1	15	6	9	11	10	0
Precision	0.93	0.89	0.65	0.77	0.79	0.75	0.86	0.81	0.68	0.89
Recall	0.93	0.82	0.77	0.81	0.91	0.83	0.84	0.81	0.73	0.86

Категория	3	14	8	7	13	4	16
Precision	0.00	0.87	0.73	0.61	0.71	0.00	1.00
Recall	0.00	0.86	0.6	0.27	0.15	0.00	0.71

3.5.3 Градиентный бустинг над решающими деревьями

Стратегия подбора параметров:

1. Фиксируем высокую скорость обучения: `learning_rate` $= 0.1$
2. При фиксированном параметре `learning_rate` определяем оптимальное число деревьев в ансамбле. В нашем случае: `n_estimators` $= 200$.
3. Настраиваем специфичные для каждого дерева параметры.

num leaves	max depth	min data in leaf
50.0	-1.0	10.0

4. Уменьшаем *learning rate* и пропорционально увеличиваем количество деревьев: `learning_rate` $= 0.01$, `n_estimators` $= 2000$

Качество классификации оптимальной модели

accuracy : 0.8050

Категория	5	12	2	1	15	6	9	11	10	0
Precision	0.93	0.92	0.65	0.78	0.79	0.75	0.87	0.85	0.67	0.89
Recall	0.94	0.84	0.79	0.81	0.92	0.81	0.82	0.79	0.75	0.84

Категория	3	14	8	7	13	4	16
Precision	0.00	0.92	0.67	0.50	0.60	0.00	0.33
Recall	0.00	0.84	0.61	0.23	0.18	0.00	0.29

3.6 Тематическая модель

Выбор модальностей и весов перед ними Исходные данные имеют гетерогенный характер, поэтому в работе предлагается использовать многомодальную модель. Модальности: текст, источник, продавец, адрес магазина.

Веса модальностей текста и класса изменяются по сетке $[1.0, 50.0, 100.0]$, а модальностей адреса, источника отзыва и имени продавца – по сетке $[0.0, 1.0, 50.0, 100.0]$.

Наилучшее качество классификации достигается на наборе:

text	class	source	market	person
1.0	100.0	1.0	1.0	1.0

Подбор количества тем в модели В данном эксперименте рассматривается мультимодальная тематическая модель, полученная на предыдущем этапе. Определяется оптимальная сложность модели – количество её тем. Сетка для перебора: от 10 тем до 1000 с шагом в 10. Функционал качества accuracy достигает оптимального значения в точке 40.

Выбор параметров регуляризации Стратегия регуляризации выстраивается исходя из общих рекомендаций и знаний предметной области:

1. *Регуляризатор декоррелирования тем.* Используется для того, чтобы сделать темы менее похожими. Для подбора оптимального коэффи-

циента производится перебор по сетке $[-10^2, 0, 10^2, 10^3, 10^4, 10^5, 10^6]$. Оптимальное значение – 10^4

2. *Регуляризатор резреженности* Φ применяется только к модальности классов, чтобы построить биективное отображение класс-тема. Оптимальное значение – -100.

Качество классификации

accuracy : 0.7416

Категория	5	12	2	1	15	6	9	11	10	0
Precision	0.79	0.98	0.61	0.66	0.76	0.69	0.79	0.63	0.72	0.77
Recall	0.92	0.70	0.70	0.82	0.88	0.79	0.73	0.74	0.51	0.76

Категория	3	14	8	7	13	4	16
Precision	0.92	0.83	0.72	0.53	0.00	0.00	0.00
Recall	0.26	0.86	0.27	0.12	0.00	0.00	0.00

По результатам работы тематической модели можно сделать интересные наблюдения:

1. Все опробованные выше классификаторы не распознают отзывы категории № 3, в то время как 92% предсказаний этого класса тематической модели оказываются верными. Тематическая модель распознаёт 26% отзывов данной категории.

2. Самые трудные для распознавания категории – 13, 4, 16. Это логично так как они содержат меньше всего примеров для обучения.

3. Если посмотреть на то, какие категории попадают в каждую из тем, то можно заметить, что в каждой теме доминирует одна категория,

то есть отображение категория-тема почти биективное.

Исключение составляют категория 5: она появляется сразу в нескольких темах; категории 11, 16 попадают в одну тему, аналогичная ситуация с категориями 13, 4, 5. Частично это объясняется тем, что в 5 категории очень много отзывов, а в 4 - очень мало.

3.7 Сравнение моделей

Класс	Метод опорных векторов		Логистическая регрессия		Градиентный бустинг		Тематическая модель	
	Precision	Recall	Precision	Pecall	Precision	Recall	Precision	Recall
5	*0.93*	*0.94*	*0.93*	*0.93*	*0.93*	*0.94*	*0.86*	*0.87*
12	0.91	0.81	0.89	0.82	**0.92**	**0.84**	0.98	0.69
2	*0.66*	*0.75*	*0.65*	*0.77*	***0.65***	***0.79***	*0.61*	*0.71*
1	0.77	0.81	0.77	0.81	**0.78**	**0.81**	0.68	0.76
15	0.79	0.91	0.79	0.91	0.79	0.92	**0.66**	**0.95**
6	0.76	0.82	**0.75**	**0.83**	0.75	0.81	0.74	0.66
9	**0.86**	**0.84**	0.86	0.84	0.87	0.82	0.63	0.82
11	**0.79**	**0.83**	0.81	0.81	0.85	0.79	0.75	0.66
10	0.68	0.74	0.68	0.73	**0.67**	**0.75**	0.78	0.38
0	***0.87***	***0.89***	*0.89*	*0.86*	*0.89*	*0.84*	*0.74*	*0.80*
3	0.0	0.0	0.00	0.00	0.0	0.0	**0.86**	**0.34**
14	**0.88**	**0.86**	0.87	0.86	0.92	0.84	0.83	0.86
8	**0.70**	**0.70**	0.73	0.66	0.67	0.61	0.69	0.52
7	0.63	0.25	0.61	0.27	0.5	0.23	**0.45**	**0.44**
13	0.83	0.29	0.71	0.15	0.6	0.18	**0.61**	**0.32**
4	0.00	0.00	0.00	0.00	0.0	0.0	**0.10**	**0.07**
16	0.83	0.71	**1.00**	**0.71**	0.33	0.29	0.50	0.71

1. Интересно, что 4-ую категорию не может выделить ни одна модель. В то время как 16-ая категория распознаётся намного лучше, хотя там меньше документов (Например, логистическая регрессия угадывает эту категорию в 100 процентах случаев).
   Этот класс содержит жалобы на работу горячей линии.
   Например:

   Почему дозвонится на вашу горячую линию невозможно?Вот уже много раз пытаюсь и безуспешно...Хочу привязать новую карту к старой утерянной.

2. Больше всего путаются категории №3("невкусно") и №2("качество продукции"). Это достаточно логично, так как "качество продукции"тоже связано со вкусом. Однако для компании различие существует:

   Например, есть пункт "раньше был лучше"и пункт "вкус не нравится". По сути это объединяет жалобы на то что покупателю не понравился купленный продукт. Но в первом случае ответственный технолог(продукт изначально был востребован, нравился а со временем перестал нравиться, значит что то в нём изменилось), во втором ответственный продакт(он решает будет тот или иной продукт продаваться в наших магазинах, и если продукт(новинка) изначально не нравиться покупателям значит это его ошибка, его ответственность. При всей схожести жалоб для покупателя, для нас это нельзя объединять.

4 Выводы

В рамках данной работы были предложены модели предобработки данных, использован новый подход для генерации дополнительных признаков с помощью тематической модели. Эти признаки помогли улучшить качество классификации базового классификатора.

Кроме того, были изучены разные методы решения задачи классификации текстов. Рассматривались метод опорных векторов, логистическая регрессия, градиентный бустинг на решающих деревьях и классификатор на основе тематических моделей. Для каждой модели подобрано оптимальное признаковое описание, настроены гиперпараметры. В работе выявлены сильные и слабые стороны каждого из подходов, проведён анализ их работы. Средний результат классификации с помощью тематической модели в данной задаче ниже, чем в случае использования стандартных методов классификации. Основными плюсами использования её в качестве классификатора является высокое качество классификации внутри отдельных классов с маленьким числом отзывов.

Работа имеет практическую пользу, так как разработанная система применяется на практике в внутри конкретной компании.

Список литературы

[1] Konstantin Vorontsov M. A. P. R. M. D., Oleksandr Frei. BigARTM: Open Source Library for Regularized Multimodal Topic Modeling of Large Collections. 2015.

[2] Vapnik V. Principles of risk minimization for learning theory // Advances in neural information processing systems. 1992. P. 831–838.

[3] Cortes C., Vapnik V. Support-Vector Networks // Machine Learning. 1995. — Sep. Vol. 20, no. 3. P. 273–297.

[4] Golub G. H., Hansen P. C., O'Leary D. P. Tikhonov regularization and total least squares // SIAM Journal on Matrix Analysis and Applications. 1999. Vol. 21, no. 1. P. 185–194.

[5] Tibshirani R. Regression Shrinkage and Selection via the Lasso // Journal of the Royal Statistical Society. Series B (Methodological). 1996. Vol. 58, no. 1. P. 267–288. URL: `http://www.jstor.org/stable/2346178`.

[6] Quinlan J. R. Induction of Decision Trees // MACH. LEARN. 1986. Vol. 1. P. 81–106.

[7] Friedman J. H. GREEDY FUNCTION APPROXIMATION: A GRADIENT BOOSTING MACHINE // The Annals of Statistics. 1999.

[8] Vorontsov K., Potapenko A. Additive regularization of topic models // Machine Learning. 2015. Vol. 101, no. 1-3. P. 303–323.

[9] Rosen-Zvi M., Griffiths T., Steyvers M., Smyth P. The Author-topic Model for Authors and Documents // Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence. 2004. P. 487–494.

[10] Rubin T. N., Chambers A., Smyth P., Steyvers M. Statistical topic models for multi-label document classification // Machine Learning. 2012. — Jul. Vol. 88, no. 1. P. 157–208.

[11] Zhou S., Li K., Liu Y. Text Categorization Based on Topic Model // International Journal of Computational Intelligence Systems. 2009. Vol. 2, no. 4. P. 398–409.

[12] Dietz L., Bickel S., Scheffer T. Unsupervised Prediction of Citation Influences // Proceedings of the 24th International Conference on Machine Learning. 2007. P. 233–240.

[13] Hospedales T., Gong S., Xiang T. Video Behaviour Mining Using a Dynamic Topic Model // Int. J. Comput. Vision. 2012. — . Vol. 98, no. 3. P. 303–323.

[14] Chong W., Blei D., Li F.-F. Simultaneous image classification and annotation // Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. 2009. P. 1903–1910.