Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»
(ФГАОУ ВО НИУ МФТИ)
Физтех-школа прикладной математики и информатики
Факультет управления и прикладной математики
Кафедра «Интеллектуальные системы»

Алексеев Василий Антонович

Поиск полного набора тем с помощью обучения нескольких моделей

010990 — Интеллектуальный анализ данных

Выпускная квалификационная работа магистра

Научный руководитель:
д. ф.-м. н. Воронцов Константин Вячеславович

Москва
2020

Содержание

Аннотация

Вероятностное тематическое моделирование — область статистического анализа текстов, где, имея на входе лишь текст, можно получить информацию о скрытых в коллекции темах как вероятностных распределениях на словах. Далее, по данной информации можно узнать, какие слова характерны для каждой темы и какие темы характерны для каждого документа. Есть много приложений тематического моделирования, но изначальная задача — исследование данных, поиск тем в коллекции текстовых документов. Одно из главных достоинств тематического моделирования в том, что решение интерпретируемо, качество тем оцениваемо человеком. Обучение при этом происходит без учителя. Но у тематических моделей есть и недостатки. Так, тематические модели неполны: одна модель не в состоянии найти все темы. Во-вторых, тематические модели неустойчивы: итоговые темы, которые выдаёт модель, зависят от начальных настроек модели, в частности от начальной инициализации матрицы принадлежностей слов темам Ф. Разные тематические модели могут выдавать разные темы, причём не все темы моделей интерпретируемые. Это приводит к тому, что исследование данных с помощью тематического моделирования обычно превращается в неорганизованный и бессистемный процесс тренировки моделей и корректировки их параметров в поисках той тематической модели, которая бы хорошо описывала данные. Принимая во внимание неполноту и нестабильность тематических моделей, в данной работе предлагается такой способ организации эксперимента по поиску тем, при котором информация о наборе данных, постепенно накапливаясь от разных моделей, используется далее при обучении новых моделей. Таким образом, информация не только не теряется, но и помогает в исследовании. Если описывать предлагаемый метод немного подробнее, то сначала происходит полный отбор интерпретируемых тем с помощью множественного обучения моделей, а затем эти отобранные темы используются для оценки качества последующих моделей. Причём с помощью автоматических способов оценки интерпретируемости тем, сравнения тем, использования неслучайной инициализации и регуляризации моделей можно как минимум ускорить и упростить процесс анализа обучаемых моделей с целью отбора тем. Предлагаемый способ оценки качества моделей основан на сравнении тем модели с теми темами, которые были отобраны заранее и которые заведомо интерпретируемые. Проведены эксперименты на нескольких коллекциях естественного языка и с несколькими тематическими моделями, демонстрирующие, что предложенный способ оценки качества моделей действительно помогает из ряда моделей выбирать ту, которая лучше всего подходит для данных. Интерпретируемость тем была оценена автоматически с помощью когерентности тем, но также возможно привлечение на этом этапе человека, что потенциально может от постановки задачи без учителя привести к постановке задачи с частичным привлечением учителя. Но это не сильно увеличит временную сложность задачи, так как, благодаря автоматическому сравнению тем модели и уже отобранных интерпретируемых тем, возможно ускорение процесса разметки новых тем пользователем.

Ключевые слова: *тематическое моделирование, множественное обучение моделей, неполнота, неустойчивость, интерпретируемость темы, когерентность темы, сравнение тем, инициализация, регуляризация.*

Введение

Актуальность темы. Тематическое моделирование — это ветвь статистического анализа текстов [1]. Если говорить конкретнее, то это автоматический способ анализа коллекции текстовых документов, нацеленный на нахождение *тем*, которые представлены в коллекции текстов. Таким образом, эти темы скрыты и заранее не известны. Более того, само понятие темы может быть может быть определено по-разному в зависимости от области и задачи. В статистическом тематическом моделировании каждая тема рассматривается как вероятностное распределение на множестве всех слов словаря. В других областях определение темы может быть основано не на частотном словаре слов. Например, в теории дискурса тема — это главный участник или главная идея на протяжении всего связного дискурса или диалога [2]. Таким образом, понятие темы может быть основано на понятии связного, осмысленного текста. Возвращаясь обратно к статистической точке зрения, тематическое моделирование даёт возможность получить сжатое представление документа в виде тем, которые затрагиваются в документе. Каждая тема характеризуется словами, по которым можно понять смысл темы. Вероятностная тематическая модель представляет каждый документ как дискретное вероятностное распределение на темах и каждую тему — как дискретное вероятностное распределение на словах. Такие модели полезны во многих областях, например в категоризации документов [3], рекомендательных системах [4], разведочном поиске [5], анализе данных социальных сетей [6]. Больше идей и возможных приложений может быть найдено в обзоре [7].

В современной литературе по тематическому моделированию представлено много моделей для разных ситуаций [7]. Самые базовые модели — это Probabilistic Latent Semantic Analysis, или PLSA [8], и Latent Dirichlet Allocation, или LDA [9]. Сотни моделей есть просто расширения PLSA и LDA. Более того, можно говорить не только о разных моделях, но и целых парадигмах создания тематических моделей. Например, байесовский подход в тематическом моделировании, начавшийся с модели LDA — это когда надо сначала описать вероятностную порождающую модель данных, затем задать априорные распределения параметров модели, и в конце получить апостериорные распределения параметров используя байесовский вывод. Другой возможный путь — аддитивная регуляризация тематических моделей, или APTM (Additive regularization for topic modelling, ARTM) [10], который основан на максимизации логарифма правдоподобия данных вместе со взвешенной суммой регуляризационных критериев, или регуляризаторов. Каждый регуляризатор представляет некоторое желаемое свойство тематической модели и, таким образом, ограничивает число возможных моделей-решений оптимизационной задачи. Однако, какую бы модель ни выбрал исследователь, каждая модель по своей природе *неполна* и *нестабильна*.

Неполнота тематических моделей означает, что нет никакой гарантии, что одна модель может в точности воспроизвести искомую скрытую тематическую структуру текстовой коллекции.

Нестабильность означает, что качество тематических моделей существенно зависит от многих вещей, связанных с процессом обучения модели. Для начала надо сказать, что идеи и гипотезы, которые используются в тематическом моделировании, в конечном итоге позволяют свести исходную задачу по поиску тем в документах к задаче матричного разложения, численное решение которой можно получить с помощью итерационного алгоритма. Однако задача матричного разложения некорректно поставлена: у неё бесконечно много решений. Также результат итерационного алгоритма зависит от инициализации модели [11]. Если несколько раз обучать тематические модели на одной и той же коллекции документов, но при разной начальной инициализации, то какие-то темы могут часто повторяться среди большого числа тематических моделей с разной инициализацией, какие-то же темы, наоборот, требуют специальной инициализации, а какие-то темы вообще могут быть неинтерпретируемыми: характеризоваться словами разных слабо связанных областей [12, 5]. Многие исследователи в своих работах оценивают стабильность тематических моделей [13, 14, 15].

Выбор лучших параметров и гиперпараметров также относится к стабильности тематических моделей. Например, изменение весов регуляризаторов в модели APTM может значительно влиять на итоговую модель [5]. Такие гиперпараметры, как число тем в модели, влияют не только на качество итоговых тем в смысле интерпретируемости, но также и на качество решения сторонних задач, где нужны векторные представления слов или документов (которые могут быть получены также и с помощью тематической модели) [16].

Как было отмечено выше, у тематического моделирования есть много приложений. В дан-

ной работе внимание сосредоточено лишь на одном: исследование данных, когда требуется найти все темы, представленные в текстовой коллекции. Таким образом, главное желание — построить такую тематическую модель, темы которой все интерпретируемы, различны и идеально описывают данные. Из-за упомянутых нестабильности и неполноты тематических моделей задача исследования данных может быть решена не до конца.

Главная идея работы заключается в том, что постепенные сбор и сохранение интерпретируемых тем с помощью множественного обучения моделей могут приводить к более качественному и полному исследованию данных с помощью тематического моделирования, чем просто в основном случайные попытки построить лучшую модель с помощью перебора параметров. На это можно смотреть как на способ проведения эксперимента. С помощью множественного обучения моделей можно получить как минимум *подмножество всех интерпретируемых тем, содержащихся в датасете*, которое начиная с этого момента и далее в работе мы будем называть *банк тем*. Само понятие банка тем будет более формально и подробно введено далее в работе, но в подходящем контексте под банком тем можно понимать просто подмножество интерпретируемых тем.

Цель работы. С целью показать, что предлагаемый способ действительно помогает в лучшем изучении данных с помощью тематического моделирования, в работе ставится следующий вопрос: существует ли такая тематическая модель, которая обеспечивает лучшее качество, рассчитанное по банку тем. Для этого надо взять несколько датасетов естественного языка, и далее для каждого датасета: создать банк тем и натренировать ряд тематических моделей. По предположению, так как все рассматриваемые датасеты в основе своей похожи (состоят их текстов естественного языка, причём тексты датасетов ни очень короткие, ни очень длинные), должна существовать такая модель среди рассматриваемых, которая бы была способна хорошо описывать большинство датасетов. Стоит отметить, что эта модель может не быть идеальной для каждого набора данных, но она должна быть лучше других рассматриваемых моделей. Если банк тем 2.4 поможет найти такую модель, то это докажет, что банк тем не просто интуитивно более простой и понятный способ проводить эксперименты с данными — он действительно помогает при оценке качества моделей, делая процесс поиска идеальной модели более простым для исследователя.

Попутно в работе ставится несколько целей:

- Проверить возможность нахождения всех тем коллекции документов с помощью методов оценки интерпретируемости тем. Так как, в виду неустойчивости и отсутствия полноты у тематических моделей, при обучении одной модели нет гарантии, что с её помощью будут найдены все темы, то выдвигается гипотеза, что при обучении *какого-то числа* моделей *каждая* тема будет найдена хотя бы одной моделью. Если это так, то существует принципиальная возможность восстановления всех тем датасета по нескольким тематическим моделям.

- Выяснить, какой эффект оказывает на итоговую модель тот факт, что часть тем при инициализации задаётся не случайно, а они берутся из отобранных заранее интерпретируемых тем 2.1. Если окажется, что качество модели заметно увеличивается, это будет значить, что, найдя хотя бы несколько хороших тем, качество всех последующих моделей можно повышать, используя эти темы при инициализации.

- Оценить, можно ли с помощью отобранных хороших тем для одной коллекции документов подбирать оптимальную стратегию обучения тематической модели для другой коллекции (стратегия — как последовательность применения регуляризаторов 2.9 [10]). Иными словами, можно ли, имея уже найденные темы для одного датасета 2.2, построить модель с как можно большим числом хороших тем для другого, нового, датасета.

Методы исследования. Для достижения поставленных целей используется аппарат статистического анализа текстов, вероятностного тематического моделирования. Для программной реализации разработанного алгоритма используется язык программирования Python.

Основные положения, выносимые на защиту.

- Предложен алгоритм создания банка тем с использованием множественного обучения моделей.
- Предложена методика оценки качества моделей с помощью банка тем.
- Разработана система для использования банка тем[1].

Научная новизна. Введено понятие полного набора тем. Предложен алгоритм по созданию полного набора тем. Предложен способ оценки качества тематической модели по подмножеству интерпретируемых тем.

Теоретическая значимость. Данное исследование вносит вклад в область статистического анализа текстов, предложенный способ оценки качества тематических моделей позволяет более полно использовать информацию об исследуемой текстовой коллекции.

Практическая значимость. Разработанные методы позволяют проводить более качественное исследование данных с помощью тематического моделирования. Результаты подтверждены экспериментально. Разработан программный модуль, реализующий алгоритм согласования сбора интерпретируемых тем и их последующего использования для оценки качества моделей.

Степень достоверности и апробация работы. Достоверность результатов подтверждена экспериментальной проверкой полученных методов на реальных коллекциях естественного языка, использованием предложенного подхода при проведении исследований, результаты которых опубликованы в рецензируемых научных изданиях.

Публикации по теме. Основные результаты по теме магистерской работы изложены в статьях, готовящихся к публикации. Также полученные результаты использованы в уже опубликованных исследованиях:

- Alekseev V. et al. *TopicNet: Making Additive Regularisation for Topic Modelling Accessible*. LREC, 2020.[2]
- Alekseev V. et al. *Topic Modelling for Extracting Behavioral Patterns from Transactions Data*. IEEE, 2019.[3]

[1] github.com/machine-intelligence-laboratory/OptimalNumberOfTopics
[2] aclweb.org/anthology/2020.lrec-1.833
[3] ieeexplore.ieee.org/abstract/document/9007329

Обозначения

Представим некоторые понятия из тематического моделирования и введём соответствующие обозначения, которые будут использоваться в работе.

- W — словарь, множество слов, которые встречаются в документах коллекции.
- D — множество документов.
- $d \in D$ — документ, который представляется последовательностью слов.
- $W_d \subseteq W$ — множество тех слов из словаря, которые встречаются в документе d.
- T — множество тем. В тематическом моделировании темы являются *скрытыми*, то есть не известны. Цель тематического моделирования — найти множество T. Каждая тема $t \in T$ описывается частотами слов и соответственно распределением этих слов в документах коллекции.
- n_{dw} — количество вхождений слова w в документ d.
- v_{wd} — частота появления слова w в документе d.

1. Мотивация и обзор области

Как уже отмечалось выше, из-за *нестабильности* и *неполноты* тематических моделей такая задача, как *исследование данных*, поиск тем в коллекции документов, может занимать много времени. Вместе с неизбежным этапом предобработки данных, исследователю приходится тренировать несколько тематических моделей, подбирать оптимальные параметры, оценивая качество итоговых тем и, возможно, сравнивая темы тренируемых моделей. Однако, даже когда наконец удаётся получить приемлемую по качеству модель, и тогда нет никакой гарантии, что её темы — это как раз те и только те темы, которые представлены в датасете, потому что тематические модели *неполны*. Существует много работ, посвящённых проблемам нестабильности и неполноты тематических моделей. С нестабильностью обычно борются путём изменений в инициализации или тренировке модели, что кажется разумным. Другая ситуация с неполнотой: предложено много эвристических путей, как можно обойти неполноту тематических моделей, иногда не совсем очевидных и понятных, но не было ещё предложено такой простой идеи, как просто поиск *всех* тем в датасете и дальнейшее использование этих тем *как есть* в качестве способа оценки качества новых тематических моделей.

Ниже представлены краткие описания некоторых релевантных работ, но ещё раз подчеркнём: преимущество предлагаемого в данной работе подхода — это простота, интерпретируемость, и принципиальная возможность вовлечения человека в процесс.

В [17] авторы предлагают тренировать несколько моделей с разными инициализациями и затем проводить кластеризацию тем всех моделей, с тем чтобы объединить похожие темы. А центры кластеров тем могут быть выбраны в качестве начальной аппроксимации тем итоговой тематической модели. В работе делается предположение, что размер кластера темы (иными словами, как часто тема была найдена разными тематическим моделями) тем больше, чем чаще тема встречается в документах текстовой коллекции.

В [18] обращается внимание на то, что слова, которые часто встречаются недалеко друг от друга в тексте, должны относиться к схожим темам. Это предположение согласуется с гипотезой о сегментной структуре текста, согласно которой слова тем распределены по тексту не случайно, вперемешку со словами других тем, а группами, сегментами [19].

Идея, представленная в работе [20] состоит в том, чтобы тренировать тематические модели несколько раз и потом проводить кластеризацию на множестве тем моделей. Если тема интерпретируемая, тогда, по мнению авторов статьи, она будет часто повторяться среди тем разных моделей и получится кластер, темы которого очень близки между собой как вероятностные распределения на словах и представляют одну тему как реальную область жизни. С другой стороны, если тема шумная, состоит из слов разных несвязаных областей, тогда кластер, соответствующий этой теме, будет разнородным. Таким образом, представленный авторами подход представляет по сути способ выявлять интерпретируемые темы, используя неустойчивость тематических моделей. Но если тема интерпретируемая и найдена только одной моделью, то предложенный подход не сможет выявить эту тему.

В работе [21] авторы хотят найти лучшую модель для коллекции документов с помощью сравнения тем одной и той же модели, но обученной на разных подвыборках документов, для вычисления *оценки качества по стабильности* для рассматриваемой тематической модели. Таким образом, из коллекции выбирается подвыборка документов (с учётом порядка), на ней обучается тематическая модель и её темы запоминаются. Это повторяется несколько раз (в статье авторы делали 10 повторов). Затем измеряется, как часто темы повторяются в моделях, обученных на разных подвыборках данных. Весь описанный выше процесс, в свою очередь, повторяется несколько раз (в статье 10 раз) и итоговая оценка качества по стабильности для модели — это медиана из полученных оценок.

В статье [22] авторы задаются целью найти *лучшую структуру модели* (в частности, число тем в модели) для модели LDA путём оценки стабильности модели. Оценка стабильности считается как среднее косинусное расстояние между всеми возможными парами тем модели. Далее, авторы предлагают способ для адаптивного нахождения лучшей структуры модели LDA с помощью кластеризации тем и оценивания стабильности модели.

В связи с тем, что нестабильность моделей связана также и с тем, что для получения численных решений используется случайная инициализация матриц слов в темах Φ и тем в документах Θ, один из возможных путей увеличения стабильности состоит в том, чтобы проводить неслучайную, *осмысленную* инициализацию — иными словами, выбирать лучшую, чем случайную, иници-

ализацию для модели.

Например, в работах [23, 24] авторы представляют понятие *якорных слов*: таких слов, которые могут служить индикаторами того, что документ относится к определённой теме; иными словами, якорные слова принадлежат только одной теме. Такое требование, что темы должны иметь якорные слова, налагает дополнительные ограничения на задачу матричного разложения. Но в [25] было показано, что задача поиска якорных слов сама по себе проще задачи матричного разложения. Якорные слова позволяют достигать локального минимума оптимизационной задачи, к которой сводится задача тематического моделирования, всего за несколько итераций. Однако стоит подчеркнуть, что не все темы в принципе могут иметь якорные слова: одного слова может быть слишком мало, чтобы описать тему, особенно если говорить о темах с большим числом слов общей лексики или о родительских и дочерних темах. Но идея использовать модель с якорными словами как начальную инициализацию для будущей модели кажется разумной, потому что такая модель содержит хотя бы *часть искомой структуры*, информации о темах коллекции. Также стоит отметить, что к определению якорных слов можно подходить по-разному. Так, можно требовать наличие только одного якорного слова у каждой темы, или можно допускать наличие больше чем одного якорного слова у тем модели.

В работе [26] авторы предлагают другое решение. Представляется понятие *контекста слова* — как слов, которые часто встречаются вместе с данным словом в тексте, недалеко друг от друга. Идея контекста слова основана на следующей гипотезе: слова, характеризующие тему, часто встречаются вместе в тексте, их относительные расположения неслучайны. Это позволяет искать начальную аппроксимацию матрицы Φ следующим образом: разбить документы исходной коллекции D на сегменты (например, параграфы или предложения). Оценить вероятности совместных встреч $p(w_1 \mid w_2)$ в тексте для всех пар слов, выбрать из всех слов словаря только те слова, которые встречаются совместно часто с *малым числом* других слов и провести кластеризацию на векторах вероятностях совместных встреч таких слов. Если каждая тема представлена в тексте в виде сегментов, то в результате кластеризации отобранные слова должны объединиться в кластеры — темы. И центр такого кластера может служить приближением темы как столбца в матрице Φ.

Далее в работе более подробно представляется понятие банка тем, описывается процесс создания банка тем с помощью множественного обучения моделей, методика оценивания качества тематической модели с использованием банка тем, и эксперименты на модельных и реальных данных.

2. Идея банка тем и полного набора тем

Определим сначала несколько понятий, которые часто используются в работе:

Определение 2.1. Везде в работе под *хорошей* темой имеется в виду интерпретируемая тема, такая, по списку самых частых слов которой человеку понятно, про что она, какую область жизни описывает.

Определение 2.2. Слова *коллекция текстовых документов*, *коллекция документов*, *датасет* в пределах работы считаются равнозначными.

Определение 2.3. Под *самыми частыми словами*, *топ словами*, *топ-словами* для темы t понимается одно и то же, а именно слова, соответствующие первым k вероятностям в отсортированном по убыванию списке $\{p(w \mid t)_w\}$. Как правило, число первых позиций k не уточняется, но под ним подразумевается некоторое очень небольшое число, по сравнению с общим количеством слов в словаре, например 10, 20, 50.

Как уже отмечалось, из-за некоторых недостатков тематических моделей, такая задача, как *исследование данных*, поиск тем в коллекции документов, может занимать много времени: помимо предобработки данных для работы моделей (от чего не уйти), приходится обучать по несколько тематических моделей, подбирать параметры, оценивая качество итоговых тем. И даже если модель получается неплохой, всё равно приходится продолжать эксперименты, потому что тематические модели не обладают полнотой: нельзя быть уверенным в том, что одна модель нашла все темы в коллекции.

В идеале хотелось бы, чтобы хватало *лишь одной модели* для поиска всех тем, чтобы за один процесс обучения можно было получить такую модель, которая была бы способна находить все темы коллекции, чтобы её темы все были интерпретируемыми и различными. К сожалению, пока это невозможно. Поэтому хочется по возможности сделать процесс тренировки моделей более организованным путём сбора интерпретируемых тем для их последующего использования. Попутно могут возникать вопросы, как упростить и ускорить, сделать более эффективным, организованным и менее затратным по времени процесс сбора хороших тем.

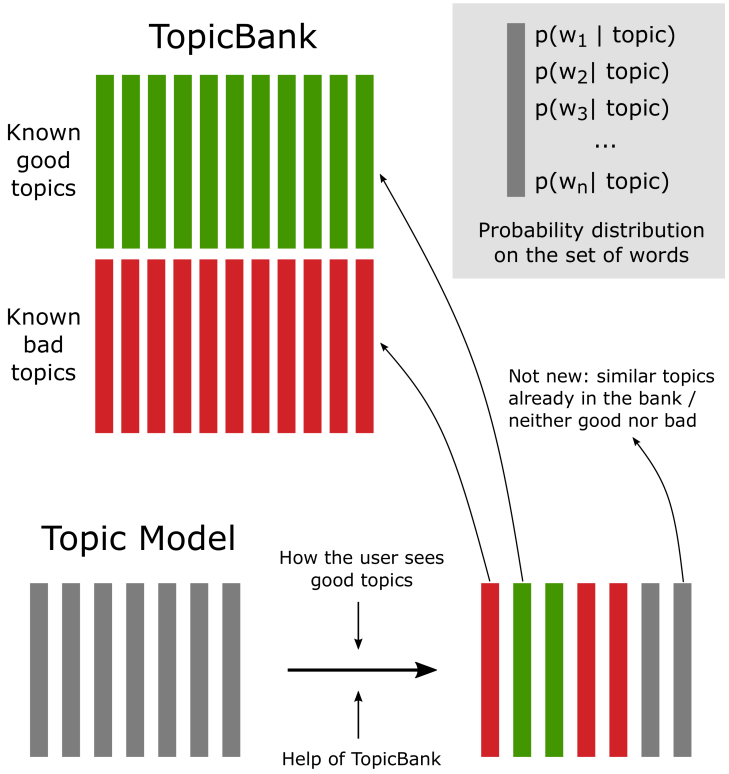


Рис. 1: Идея банка тем: в нём накапливаются интерпретируемые темы и (опционально) неинтерпретируемые. Банку для функционирования требуется информация от человека о том, как определять интерпретируемые темы: это отображение из темы в булевское значение, означающее, интерпретируема тема или нет. Также банк должен уметь сравнивать две темы между собой, с целью выявления среди новых тем тех, которые на самом деле уже похожи на некоторые темы, находящиеся в банке. Это отображение, которое переводит две темы в булевское значение, означающее, близки две темы или нет.

Для более удобной работы с темами моделей предлагается использовать *банк тем*: интерпретируемые темы постепенно собираются в банке. Главная задача банка тем — это сбор интерпретируемых тем с их последующим использованием для оценки качества последующих тематических моделей.

В идеале, банк тем из таких интерпретируемых тем, которые совместно образуют *полный набор тем*. Поясним подробнее, что мы понимаем под полным набором тем. Не столько в форме определения, сколько в форме свойств, которые ожидаются от тем полного набора.

Определение 2.4. Темы из полного набора тем

- интерпретируемы
- различны (точнее, не существует линейной зависимости между темами)
- совместно образуют такое матричное разложение $\Phi \cdot \Theta$, которое максимизирует правдоподобие коллекции (иными словами, вместе темы представляют такую модель, которая хорошо описывает данные)

В данной работе мы будем требовать от банка тем лишь два свойства: интерпретируемость и различность тем. То есть будем считать, что допустимо, когда некоторые интерпретируемые темы есть в датасете, но отсутствуют в соответствующем банке тем.

Даже если процесс обучения моделей удалось спланировать таким образом, что получающиеся тематические модели интерпретируемы, то из-за неполноты моделей человеку всё равно может понадобиться много раз тренировать модели и выбирать среди их тем интерпретируемые, из которых можно будет получить полный набор тем. Некоторые из тем моделей могут быть неинтерпретируемыми. Более того, темы могут повторяться от модели к модели (как интерпретируемые, так и нет). В данной работе делается предположение, что *с помощью множественного обучения моделей возможно собрать полный набор тем* (в смысле определения, данного выше).

Банк тем служит своего рода *обёрткой* над тематическим моделированием, позволяющей уменьшать число моделей, требуемое для получения полного набора тем: информация о датасете накапливается и помогает выбирать лучшие модели 1.

Постепенный отбор интерпретируемых тем также может помочь в задаче определения числа тем в коллекции: когда в банк тем уже нельзя добавить тему, увеличивающую правдоподобие банка тем. Даже если тема интерпретируемая, но является дубликатом одной из интерпретируемых тем, уже находящихся в банке, тогда её добавление в банк не позволит построить модель, обеспечивающую более высокое правдоподобие (модель, темы которой — это именно те темы, которые хранятся в банке тем). Таким образом, танк тем может помогать как в поиске интерпретируемых тем, так попутно в определении их числа.

Для полноты картины, дадим два более формальных определения банка тем. Сначала — как функционального объекта, инструмента для исследователя.

Определение 2.5 (Банк тем как функциональный объект)**.** Банк тем — инструмент, способ работы с тематическими моделями, основные функции которого:

- хранение информации о просмотренных темах (вероятности слов; оценка качества темы; некоторые выявленные зависимости между темами, такие как похожесть, родственность)
- ускорение анализа моделей путём сортировки новых тем в порядке непохожести до ранее сохранённых тем (то есть от более новых к, вероятно, похожим на ранее просмотренные темы)

И более математическое определение сущности банка тем.

Определение 2.6 (Банк тем как математический объект)**.** О банке тем можно думать как о кортеже

$$B = \langle T, \rho \rangle$$

где T — множество тем, $\rho : T \times T \to \mathbb{R}_+$ — функция расстояния между темами.

При этом определены операции добавления в банк новой темы:

$$\text{add} : \langle T, \rho \rangle, t \mapsto \langle T \cup \{t\}, \rho \rangle$$

И поиска ближайшей к данной теме темы из банка:

$$\text{nearest} : \langle T, \rho \rangle, t \mapsto \arg\min_{t_1 \in T} \rho(t_1, t)$$

Для банка тем как функционального объекта можно привести примеры конкретных сценариев, где он может использоваться.

- Сохранить тему, найденную какой-нибудь моделью.
- Показать темы, похожие на данную. Показать дубликаты/родителей/детей данной темы банка.
- Провести анализ тем модели: вывести темы в порядке удаления от тем, сохранённых в банке (показывать новые темы в первую очередь). Для каждой темы вывести информацию о её топ-словах, о покрытии коллекции темой. Пользователь имеет возможность сохранить тему в банк как хорошую/плохую; указать, дубликат ли тема самой близкой темы в банке; указать, родительская ли тема или дочерняя по отношению к ближайшей теме банка.

- Если пользователь имеет в распоряжении темы, которые он считает хорошими, то он может посчитать для этих тем некоторые оценки качества (когерентность/чистота/контраст/...). И далее при обучении новых моделей можно подавать их банку тем для обработки в автоматическом режиме (то есть банк не будет задавать вопросов относительно тем), указав те значения функции качества темы, которые соответствуют хорошим темам.

2.1. Задача восстановления полного набора тем

Сформулируем задачу восстановления полного набора тем коллекции документов D.

Пусть даны D — коллекция документов и $\Phi_{original}$ — соответствие между темами и словами, известная разметка тем по словам. За Θ обозначим соответствие между темами и документами (одна тема может соответствовать более чем одному документу, и один документ может быть причастен нескольким темам), которое получается с помощью тематической модели. Также тематическая модель выдаёт как результат матрицу Φ. Обозначим за $\text{seed} \in \mathbb{R}$ случайность в начальной настройке модели (например, инициализация матрицы Φ_0 модели) или в процессе получения численного решения (например, порядок просмотра документов в методе простой итерации или подмножество документов, используемое для обучения).

Пусть при заданной коллекции D документов и фиксированной модели m из множества $\mathscr{M}$ моделей $M : \text{seed} \mapsto \langle \Phi, \Theta \rangle$ — процесс обучения модели (EM алгоритм для решения задачи максимизации правдоподобия коллекции $L(\Phi, \Theta) \to \max_{\Phi,\Theta}$), где Φ — темы, полученные с помощью тематической модели, обученной при начальной инициализации матрицы $\Phi_0(\text{seed})$.

Пусть $\Phi_{interpretable}(\text{seed} \mid q)$ — интерпретируемые (по функционалу качества тем q) темы из $\Phi(\text{seed})$. И задача ставится так, чтобы построить процесс обучения $\mathscr{M}$ таким образом, чтобы все темы были найдены:

$$\bigcup_{i}^{K} \Phi_{interpretable}(\text{seed}_i) \supseteq \Phi_{original}$$

При этом также важно, чтобы K — число моделей — было как можно меньшим, и чтобы не было лишних тем:

$$\bigcup_{i}^{K} \Phi_{interpretable}(\text{seed}_i) \subseteq \Phi_{original}$$

2.2. Инициализация моделей, повышающая качество итоговых тем

Так как неустойчивость вызвана случайной инициализацией, то один из способов повысить устойчивость — сделать *осмысленную инициализацию*, подобрать лучшее, чем случайное, начальное приближение для модели.

Например, в работах [23, 24] вводится понятие *якорных слов*: таких слов, о которых можно сразу судить о принадлежности документа к определённой теме, то есть якорные слова принадлежат только к одной теме. Требование наличия якорных слов у тем накладывает дополнительные ограничения на задачу матричного разложения, решаемую в тематическом моделировании. Но показано [25], что задача поиска якорных слов тем решается быстрее самой задачи матричного разложения. А при найденных якорных словах, достаточно всего нескольких итераций для достижения локального минимума оптимизационной задачи. Но не у всех тем могут быть якорные слова: одного слова может быть недостаточно для определения темы, особенно если речь идёт о темах с большим количеством слов общей лексики или о родительских и дочерних темах. Применять же такую модель в качестве начальной *инициализации* матрицы Φ при решении задачи матричного разложения кажется имеющим смысл, ведь такие матрицы будут содержать в себе по крайней мере часть искомой структуры, информации о темах коллекции. Стоит отметить ещё, что к определению якорного слова можно подходить по-разному. Так, можно считать, что у каждой темы может быть лишь одно якорное слово или что их может быть несколько.

Определение 2.7. Везде в работе под словом *Arora* понимается алгоритм поиска начального приближения матрицы Φ по алгоритму из [23]

В работе [26] предлагается другой подход. Вводится понятие *контекста* слова, как слов, которые часто встречаются с данным словом недалеко друг от друга в тексте коллекции. За таким понятием стоит следующая гипотеза: слова, которые наиболее точно характеризуют тему, встречаются в тексте как правило вместе, а не вразброс. Это позволяет искать начальное приближение Φ следующим образом: можно разбить документы исходной коллекции D на сегменты (например, абзацы или предложения). Оценить по ним вероятности совместных близких встреч $p(w_1 \mid\mid w_2)$ в тексте для всех пар слов, выделить среди всех слов те, которые *встречаются с небольшим количеством других слов достаточно часто*. И провести кластеризацию векторов вероятностей совместных встреч таких слов. Если каждая тема представлена в тексте сегментами, то в результате кластеризации отобранные слова должны объединиться в один кластер — искомую тему. И центр этого кластера можно положить приближением темы как столбца в матрице Φ.

Определение 2.8. Далее в работе под словом *CDC* (Contextual Document Clustering) имеется в виду способ инициализации Φ с помощью алгоритма из [26].

С приведёнными способами инициализации матрицы Φ будет сравниваться способ инициализации с помощью части тем из банка тем. В работе не ставилось целью провести как можно более обширное сравнение существующих методов инициализации Φ. Поэтому для сравнения с инициализацией по банку тем были выбраны лишь два других.

2.3. Целенаправленный поиск новых тем

Существуют работы [27, 28, 29], в которых авторы задаются целью поиска новых тем в новостных потоках. В данной же работе интересует поиск тем *не* в изменяющихся со временем коллекциях, таких как новостные потоки, а поиск новых тем в том же самом статичном датасете, но при условии, что *какая-то часть тем уже найдены*.

Кажется, что накопленное знание должно помогать в поиске новых тем: либо при обучении модели, либо при инициализации модели перед обучением. Второй подход рассматривается в данной работе. Предлагается искать документы, которые модель не может отнести ни к одной теме из уже найденных. Либо эти документы относятся к темам, которые модель не смогла найти, либо содержание документов слишком общее и не поддаётся тематизации. Можно дополнительно предпринимать попытки отбора среди этих документов тех, в которых с большей вероятностью могут скрываться новые темы. На отобранных плохо тематизированных документах можно строить новую тематическую модель, в надежде, что ей будет проще искать темы.

2.4. Использование другой, уже размеченной, коллекции при анализе новой

Определение 2.9. Пусть $\mathscr{R}$ — множество регуляризаторов, которые могут применяться при обучении моделей. Под *стратегией* обучения s будем понимать последовательность

$$s = \{A_i\}_{i=1}^{N}, \quad A_i \in 2^{\mathscr{R}}$$

где N — число итераций обучения модели (количество обновлений матриц Φ, Θ), а A_i — множество *активных* на итерации i регуляризаторов, то есть тех, которые принимают участие в обучении модели на итерации i.

Из всего множества стратегий можно выбрать лучшую по тому, какого качества получается итоговая модель. Выбор стратегии, таким образом, зависит от способа оценки качества модели и от самого датасета, на котором проходит обучение. В работе далее хочется сравнить лучшие стратегии обучения для моделей на разных датасетах при разных способах оценки качества моделей, один из которых — с помощью отобранных заранее хороших тем. Если окажется возможным подбирать стратегию по такому критерию, то это будет значить, что, изучив одну коллекцию и найдя её темы, можно использовать это знание для построения хороших моделей на другой коллекции, путём выбора лучших стратегий для обучения.

3. Тематическое моделирование

В данном разделе мы представим более подробно аппарат тематического моделирования, применяемый в работе, и введём нужные обозначения.

Обозначим за Dколлекцию текстовых документов, за W — множество всех слов (словарь). Термин из W может быть отдельным словом или сочетанием слов. Каждый документ $d \in D$, таким образом, может быть представлен как упорядоченная последовательность n_d терминов из W. Пусть n_{dw} обозначает количество раз, сколько термин $w \in W$ появляется в документе $d \in D$.

Теперь предположим, что каждый термин в каждом документе соответствует некоторой скрытой теме из конечного множества тем T. Другими словами, о текстовой коллекции можно думать как о наборе троек $\{(w_i, d_i, t_i)\}_{i=1}^{n}$ полученных независимо из дискретного распределения $p(w, d, t)$ над конечным множеством $W \times D \times T$. Отметим ещё раз, что слова и документы есть видимые переменные, в то время как темы — скрыты.

В тематическом моделировании обычной является практика принятия гипотезы *мешка слов*: когда каждый документ рассматривается как мультимножество слов, то есть без учёта порядка слов. Считается, что для нахождения тем порядок слов в документах не важен. Хотя такое предположение и кажется сильным упрощением действительности, в нём есть смысл, потому что, например, обе последовательности слов “Джек Лондон американский писатель автор известных приключенческих романов” и “автор Лондон известных романов Джек писатель приключенческих американский” — хоть и отличаются друг от друга последовательностью слов, но основная тема — литература — в обоих примерах угадывается верно.

В тематическом моделировании также принимается гипотеза *условной независимости*, которая гласит, что слово относится к теме в не зависимости от того, в каком документе находится слово: $p(w \mid d, t) = p(w \mid t)$. Вероятностные тематические модели описывают процесс порождения документов через смесь распределений $\phi_{wt} \equiv p(w \mid t)$ и $\theta_{td} \equiv p(t \mid d)$. И при допущении условной независимости это можно описать так

$$p(w \mid d) = \sum_t p(w \mid t) p(t \mid d) = \sum_t \phi_{wt} \theta_{td}$$

Таким образом, о теме в тематическом моделировании можно думать как о распределении на множестве слов $p(w \mid t)$, $w \in W$. Более того, каждая тема также характеризуется её распределением на множестве документов: $p(t \mid d)$, $d \in D$, с помощью которого можно, например, оценивать качество темы, насколько она понятна человеку. Тематическое моделирование, в сущности, сводится к поиску распределений ϕ_{wt} и θ_{td} по коллекции D. Эта задача эквивалентна поиску решения задачи матричного разложения матрицы известных частот слов в документах $\left(\frac{n_{wd}}{n_d}\right)_{W \times D}$ на произведение матриц $\Phi \cdot \Theta$, где

$$\Phi \equiv (\phi(w \mid t))_{W \times T} \equiv (\phi_{wt})_{W \times T}$$

$$\Theta \equiv (\theta(t \mid d))_{T \times D} \equiv (\theta_{td})_{T \times D}$$

Все упомянутые выше матрицы стохастические: их колонки неотрицательны, нормированы и представляют дискретные вероятностные распределения.

Существуют разные виды тематических моделей. Одна из самых первых, и в то же время самых простых и понятных — модель PLSA [30] — получается максимизацией правдоподобия коллекции $p(\Phi, \Theta)$

$$p(\Phi, \Theta \mid D) = \prod_{d \in D} \prod_{w \in W_d} p(d, w)^{n_{dw}} = \prod_{d \in D} \prod_{w \in W_d} p(w \mid d)^{n_{dw}} p(d)^{n_{dw}} \to \max_{\Phi, \Theta}$$

И, если перейти от правдоподобия к его логарифму, то задача примет вид

$$\mathscr{L}(\Phi, \Theta \mid D) \equiv \ln p(\Phi, \Theta \mid D) = \sum_{d \in D} \sum_{w \in W_d} n_{wd} \log p(w \mid d) = \sum_{d \in D} \sum_{w \in W_d} n_{wd} \log \sum_t \phi_{wt} \theta_{td} \to \max_{\Phi, \Theta}$$

с линейными ограничениями $\sum_{w \in W} \phi_{wt} = 1,\ \phi_{wt} \geq 0$ и $\sum_{t \in T} \theta_{td} = 1,\ \theta_{td} \geq 0$.

Точка локального экстремума сформулированной задачи удовлетворяет системе уравнений, которая может быть численно решена итеративными методами, с обновлениями матриц Φ и Θ

[31] на каждой итерации. Но при таком способе решения необходимо *инициализировать* матрицы Φ и Θ — поэтому способ удачной инициализации есть отдельный вопрос (несколько подходов будут рассмотрены далее в работе).

В оптимизируемый функционал можно включать члены, которые позволяют получать как результат темы, удовлетворяющие некоторым дополнительным требованиям. В работах [32, 31, 10] предложен подход к обучению тематических моделей, названный аддитивной регуляризацией тематических моделей (ARTM: Additive Regularization of Topic Models). Регуляризаторы позволяют сокращать допустимое множество решений задачи матричного разложения до тех решений, которые удовлетворяют некоторым свойствам. Например, с помощью регуляризаторов можно требовать модели, чтобы её темы были различны, чтобы у каждой темы было лишь небольшое число наиболее характерных для неё слов или, наоборот, чтобы как можно больше слов часто встречались в теме (таким образом можно строить фоновые темы, содержащие слова общей лексики). Регуляризация используется для получения решения с заданными свойствами, и должна служить также и для повышения устойчивости тем модели. Как говорилось выше, APTM подход состоит в введении дополнительных ограничений на решения оптимизационной задачи путём прибавления в логарифму правдоподобия дополнительных регуляризационных членов с неотрицательными весами τ_i:

$$\mathscr{L}(\Phi, \Theta) + \sum_{i=1}^{n} \tau_i R_i(\Phi, \Theta) \to \max_{\Phi,\Theta}$$

Условия Каруша-Куна-Таккера дают необходимые условия для локального минимума в виде системы уравнений. Эта система может быть численно решена, например, методом простой итерации, который в данном случае будет эквивалентен EM алгоритму [10]. Сила APTM подхода в том, что каждое регуляризационное слагаемое в итоге выражается в виде дополнительной линейной добавки на M-шаге.

Иерархическое тематическое моделирование — это подраздел тематического моделирования, где строятся многоуровневые тематические модели, когда каждая *родительская тема* из T представляется как смесь *дочерних тем* S, где $|S| > |T|$ [33]:

$$\underbrace{p(w \mid t)}_{\phi_{wt}^{parent}} = \sum_{s \in S} \underbrace{p(w \mid s)}_{\phi_{ws}^{child}} \underbrace{p(s \mid t)}_{\psi_{st}} \tag{1}$$

Другими словами, $\Phi_{W\times T}^{parent} = \Phi_{W\times S}^{child} \Psi_{S\times T}$, где $\Psi_{S\times T}$ есть матрица, представляющая связи между темами двух уровней (родительскими и дочерними). Если переформулировать эту связь между темами в нотации APTM, то окажется, что родительские темы могут рассматриваться как псевдодокументы с частотами слов, равными n_{wt}.

3.1. Регуляризаторы

Сглаживание и разреживание.

Темы T тематической модели могут быть поделены на два вида: $T = S \sqcup B$, где S есть предметные темы и B — фоновые темы.

Предметные темы S специализированы и состоят преимущественно из слов, относящихся к определённой, специализированной области. Такие темы разрежены и слабо скоррелированы между собой. Разреживающий регуляризатор основан на максимизации KL-дивергенции между распределениями ϕ_{wt}, θ_{td} и соответствующими равномерными распределениями.

Фоновые темы B состоят из слов общей лексики и равномерно распределены по документам коллекции. Сглаживающий регуляризатор основан на минимизации KL-дивергенции между распределениями ϕ_{wt}, θ_{td} и соответствующими равномерными распределениями.

Выражение для разреживающего регуляризатора

$$R(\Phi, \Theta) = -\beta_0 \sum_{t \in S} \sum_{w \in W} \beta_w \ln \phi_{wt} - \alpha_0 \sum_{d \in D} \sum_{t \in S} \alpha_t \ln \theta_{td} \to \max$$

И для сглаживающего

$$R(\Phi, \Theta) = \beta_0 \sum_{t \in B} \sum_{w \in W} \beta_w \ln \phi_{wt} + \alpha_0 \sum_{d \in D} \sum_{t \in B} \alpha_t \ln \theta_{td} \to \max$$

Где β_0 и α_0 есть неотрицательные веса, которые подбираются экспериментально.

Декорреляция.
Регуляризатор декоррелирования увеличивает расстояние между темами.

$$R(\Phi) = -\frac{\tau}{2} \sum_{t \in T} \sum_{s \in T \setminus t} \sum_{w \in W} \phi_{wt} \phi_{ws} \to \max$$

Так как регуляризатор зависит только от матрицы Ф слов в темах, то эффект от регуляризатора будет проявляться только на М-шаге при обновлении матрицы Ф.

Отбор тем.
Регуляризатор по отбору тем максимизирует KL-дивергенцию между $p(t) = \sum_d p(d)\theta_{td}$ и равномерным распределением по темам:

$$R(\Theta) = \frac{n}{|T|} \sum_{t \in T} \ln \sum_{d \in D} p(d)\theta_{td} \to \max$$

Таким образом, применение регуляризатора приводит к тому, что некоторые темы могут исчезнуть из матрицы Θ, то есть им будут соответствовать нулевые строки.

4. Способы оценки качества тем

Оценка качества тем и тематических моделей не простая задача. В идеале, каждая тема должна быть просмотрена человеком: распределение темы на словах и то, как тема распределена в документах. Очевидно, это невозможно, так как требует больших затрат времени и сил. Поэтому есть автоматические способы оценки качества моделей. Они же могут быть внутренними (intrinsic) и внешними (extrinsic). Внутренние означают, что качество оценивается на том же датасете, на котором происходило обучение модели. Внешние же про то, чтобы использовать для оценки качества другие датасеты или же использовать тематические модели для решения других, внешних задач с известной разметкой. Ниже представлены некоторые внутренние критерии качества тем и моделей, которые и будут использоваться в экспериментах, описанных в следующих разделах в данной работе.

Как способ оценить качество всей модели в целом, можно использовать перплексию, которая тесно связана с правдоподобием $\mathscr{L}(\Phi, \Theta)$

$$\text{Perplexity}(\Phi, \Theta) = e^{-\mathscr{L}(\Phi,\Theta)}$$

Чем выше правдоподобие модели, тем ниже перплексия, и наоборот.

Есть также и способы оценки качества отдельных тем. Например, чистота $\text{Purity}(t \mid \text{threshold}) = \sum_{w \in W_t} p(w \mid t)$ и контраст $\text{Contrast}(t \mid \text{threshold}) = \sum_{w \in W_t} \frac{1}{|W_t|} p(t \mid w)$, где $W_t = \{w \in W \mid p(t \mid w) > \text{threshold}\}$ — это ядро темы, оценивают качество темы на основе информации в матрице Ф [10]. Естественный способ выбрать порог threshold — это положить его равным $1/|W|$: так чтобы ядро состояло только из таких слов, вероятность которых для данной темы выше, чем равномерная [34].

Отметим однако, что в данной работе, для сбора полного набора тем текстовой коллекции, нас будут интересовать именно способы оценки качества темы, а не всей модели целиком.

В работах [35, 12, 36] авторы предлагают метод для оценки качества тем, названный *когерентностью*: когда решение о качестве темы выносится на основе того, как часто самые вероятные слова темы встречаются парами недалеко друг от друга в тексте (в сравнении с числом раз, когда одно и другое слова просто встречались в тексте, неважно, близко друг к другу или нет). Математическое выражение представленной идеи следующее:

$$\text{Coherence}(t \mid D, k) = \underset{w_i, w_j \in \text{top}(t,k)}{\text{Average}} \frac{p(w_i, w_j)}{p(w_i)p(w_j)}$$

где $p(w_i, w_j)$, $p(w_i)$ — вероятности встретить слово w_i или два слова w_i, w_j внутри окна некоторого размера в тексте, и $\text{top}(t, k)$ — это первые k топ-слов (самых частых слов) темы t. Вероятности оцениваются по известным частотам слов в документах. Вообще, когерентность — это попытка автоматизировать человеческий способ оценки качества темы. Как человек может понять, хорошая тема или нет? Естественный способ заключить, что тема хорошая — это в том случае, если она *интерпретируемая*, если по составляющим её словам (и по распределению этих слов в документах коллекции) можно понять, что это за тема, описывает ли она реальную область жизни. В упомянутых выше работах авторы как раз показали, что такая когерентность хорошо коррелирует с оценками тем, полученными с привлечением человека.

В статье [19] авторы предлагают немного другой подход к оценке качества тем, названный *внутритекстовой когерентностью*, учитывающий распределение всех слов темы по документам корпуса, а не только самых частых слов темы. Такая идея основана на гипотезе о сегментной структуре текста естественного языка: когда темы представлены в тексте как сегменты, а не как слова, расположенные вразнобой. У метода оценки качества, основанного лишь на небольшом количестве самых частых слов темы, есть привлекательная сторона: например, скорость вычислений (при внутритекстовом подходе для вычисления качества новой модели надо каждый раз просматривать целиком всю коллекцию, метод же по топ-словам может обходиться лишь одним проходом по коллекции). Недостаток подхода, основанного на топ-словах, состоит в том, что сведение темы лишь к малому числу слов приводит к тому, что большое количество информации о теме вообще не используется при оценке качества. Разница между внутритекстовом подходе к вычислению когерентности и по топ-словам более подробна рассмотрена в [19].

5. Эксперименты

5.1. На модельных данных

Идея в том, чтобы проверить, что возможно восстановить все изначальные темы датасета с помощью обучения нескольких моделей.

Постановка эксперимента: синтетический датасет, обучение моделей.
Датасет создавался таким образом, чтобы в нём было 10 тем, и в каждой по 10 слов с равными вероятностями. У тем не было общих слов (слов, которые с ненулевыми вероятностями входят в несколько тем). По каждой теме создавалось 100 документов (вероятность основной темы в документе 0.8 и ещё несколько других). Далее, несколько раз обучалась модель на 5 темах в течение 10 итераций с некоторой начальной инициализацией.

Результаты.
Тема считалась найденной моделью, если 7 из 10 топ слов и первые 2 топ слова темы оказывались правильно угаданными. Порядок слов не учитывался, потому что слова в синтетических темах были равновероятными.

В итоге после 24 обучений моделей, *все исходные темы были найдены* (найдены в смысле, оговорённом выше). Таким образом, продемонстрирована потенциальная возможность получения с помощью тематических моделей полного набора тем, затрагиваемых в коллекции документов.

Более того, темы находились с разными частотами: одна тема была найдена 22 моделями, несколько других были найдены только одной моделью. В среднем одна тема была найдена 6 моделями. И из 5 тем каждой модели в среднем две оказывались хорошими.

5.2. На реальных данных

5.2.1. Описание данных

Датасет, который используется во всех экспериментах (и по созданию банка тем, и по его использованию для оценки качества моделей) — это коллекция научно-популярных статей «Постнаука»[4]: 3446 документов, несколько десятков тем (19) 1. Число тем примерно известно, потому

[4] postnauka.ru

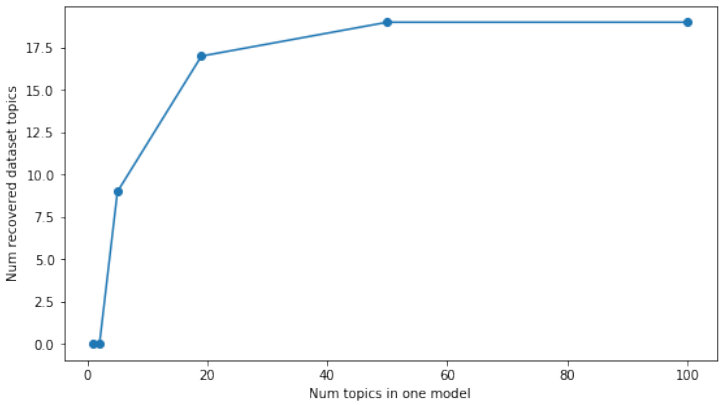


Рис. 2: Зависимость числа тем, которые удалось восстановить, от числа тем в одной тематической модели из серии моделей. Чем больше тем в модели, тем больше тем удаётся найти.

что статьи изначально были поделены по секциям. Исходные темы — с разным числом документов (коллекция несбалансирована). Для всех тем известны их оценки качества: когерентности, чистота, контраст.

Везде, где не оговорено специально, предполагается, что эксперименты выполнялись на данных коллекции «ПостНаука».

Также для проведения экспериментов было выбрано несколько других датасетов. Все датасеты представлены в таблице 2: упомянутая ранее ПостНаука[5], Reuters[37], Brown[6] [38], Twenty Newsgroups (20 NG)[7] [39], AG News [40], Watan2004 [41], Хабрахабр[8], и русская Википедия[9]. Главным критерием отбора датасетов был следующий: естественный язык (не важно, какой) и по крайней мере несколько чётко выраженных, различных тем.

5.2.2. Зависимость необходимого числа обучаемых моделей от числа тем в одной модели

Цель данного эксперимента — проверить, как зависит число обучаемых моделей, нужное для получения всех тем, от числа тем в одной тренируемой модели (число тем фиксировано и одинаково во всех моделях). В данном эксперименте происходило следующее. Фиксировалось число тем, обучались тематические модели с данным числом тем и разными начальными инициализациями. Зрительно анализировались списки топ-слов тем. Если по списку топ-слов темы ей можно было дать понятное название, и тема с таким названием была в датасете, то тема считалась найденной моделью.

Если в модели тем примерно столько же, сколько и в датасете, то результат восстановления тем не идеальный: тем в датасете было 19, и понадобилось 20 раз обучать модель на 19 темах, чтобы восстановить 16 тем (рисунок 2).

Если обучать модели на меньшем числе тем, чем было в датасете, то результаты ещё хуже. Например, крайний случай, когда строились модели на 1-2 темах: темы ожидаемо были низкого качества, и повторялись на разных инициализациях.

Но и если взять тем чуть меньше, чем в датасете, например 5 тем, то всего за 20 попыток получилось найти 8 тем: при этом какие-то часто повторялись, часто были смешанные неинтерпретируемые темы, какие-то темы состояли из двух интерпретируемых тем (таблица 3), какие-то темы удавалось различать, но в списке топ-слов некоторые слова были не относящимися к теме (таблица 4).

Если обучать модели на заведомо большем числе тем, чем есть в датасете, например 50 или 100 (в датасете 19), то тут получилось восстановить все темы за 6 (3 в случае 100 тем) проходов. При этом часто темы восстанавливались на глаз почти идеально 5, было найдено много явных

[5] postnauka.ru
[6] Reuters и Brown датасеты могут быть скачаны с помощью NTLK: nltk.org/book/ch02.html
[7] Датасет может быть скачан с помощью Scikit-learn: scikit-learn.org/0.19/datasets/twenty_newsgroups.html
[8] habr.com
[9] dumps.wikimedia.org/ruwiki

Математика	Технологии	Физика	Химия
математика (1.57)	технология (1.51)	частица (2.74)	химия (2.12)
задача (0.84)	робот (1.21)	электрон (1.53)	молекула (1.87)
декарт (0.79)	сеть (1.04)	кварк (1.51)	материал (1.61)
математический (0.73)	компьютер (0.97)	атом (1.28)	соединение (1.30)
математик (0.73)	использовать (0.82)	энергия (1.16)	вещество (1.24)
решение (0.66)	дать (0.80)	вселенная (1.09)	структура (1.13)
точка (0.63)	устройство (0.72)	фотон (1.05)	элемент (0.95)
функция (0.63)	работать (0.63)	физика (0.93)	полимер (0.86)
модель (0.59)	задача (0.61)	физик (0.90)	метод (0.85)
переменный (0.57)	программа (0.53)	эксперимент (0.88)	атом (0.80)

Земля	Астрономия	Биология	Медицина
земля (2.92)	звезда (3.87)	клетка (2.65)	пациент (1.61)
планета (1.88)	галактика (3.12)	ген (1.75)	препарат (1.22)
атмосфера (1.16)	вселенная (1.94)	организм (1.10)	заболевание (1.16)
вулкан (1.10)	солнце (1.28)	мозг (1.03)	врач (1.08)
планет (1.09)	вещество (1.22)	днк (1.02)	болезнь (1.05)
лёд (0.94)	планета (1.12)	животное (1.01)	клетка (1.05)
километр (0.92)	земля (1.07)	белка (0.86)	лечение (0.88)
ледник (0.88)	объект (1.04)	растение (0.80)	медицина (0.83)
температура (0.79)	масса (1.04)	бактерия (0.76)	организм (0.76)
океан (0.75)	телескоп (0.96)	геном (0.68)	сон (0.71)

Психология	Экономика	История	Политика
ребёнок (1.41)	экономика (1.59)	история (1.03)	государство (1.43)
психология (0.93)	страна (1.01)	историк (0.70)	политика (1.23)
мозг (0.88)	цена (0.82)	власть (0.62)	политический (1.13)
психолог (0.83)	экономист (0.79)	народ (0.50)	власть (1.10)
задача (0.79)	компания (0.77)	война (0.48)	война (0.97)
память (0.76)	китай (0.76)	король (0.47)	демократия (0.92)
внимание (0.70)	экономический (0.71)	государство (0.47)	страна (0.88)
испытуемый (0.66)	рынок (0.67)	сталин (0.43)	свобода (0.67)
объект (0.59)	деньга (0.63)	политический (0.39)	партия (0.60)
мышление (0.59)	кризис (0.49)	революция (0.36)	гражданин (0.56)

Социология	Культура	Образование	Язык
социология (1.28)	культура (1.48)	университет (2.10)	язык (7.71)
социолог (0.87)	фильм (0.74)	образование (1.42)	слово (3.71)
социальный (0.84)	искусство (0.62)	школа (1.33)	словарь (1.13)
общество (0.79)	музей (0.54)	студент (1.25)	лингвист (1.00)
объект (0.60)	культурный (0.51)	наука (1.06)	глагол (0.81)
сообщество (0.59)	книга (0.48)	вуз (0.65)	предложение (0.80)
отношение (0.57)	текст (0.45)	преподаватель (0.58)	текст (0.74)
пространство (0.55)	советский (0.44)	учитель (0.52)	русский (0.71)
событие (0.53)	кино (0.44)	должный (0.51)	говорить (0.70)
город (0.50)	традиция (0.40)	школьник (0.50)	звук (0.70)

Философия	Религия	Россия
философия (1.75)	бог (1.49)	россия (2.79)
философ (1.27)	святилище (1.00)	страна (0.92)
философский (0.81)	религия (0.72)	русский (0.90)
понятие (0.67)	имя (0.65)	государство (0.74)
свобода (0.62)	царь (0.61)	российский (0.61)
платон (0.51)	традиция (0.55)	отношение (0.42)
книга (0.51)	христианство (0.54)	проект (0.35)
знание (0.49)	религиозный (0.47)	москва (0.35)
аристотель (0.44)	божество (0.46)	реформа (0.29)
смысл (0.43)	церковь (0.45)	толстой (0.29)

Таблица 1: Темы датасета «ПостНаука»

Название	$\|D\|$	Язык
ПостНаука	3 446	русский
Reuters	10 788	английский
Brown	500	английский
20 NG	18 846	английский
AG News	127 600	английский
Watan2004	20 291	арабский
Хабрахабр	133 978	русский
Википедия	1 255 836	русский

Таблица 2: Датасеты, использующиеся в экспериментах ($|D|$ означает число документов в датасете).

Физика-Астрономия	История-Социология	Наука-Астрономия
частица (0.66)	человек (0.63)	наука (0.61)
теория (0.55)	социальный (0.53)	человек (0.60)
звезда (0.55)	другой (0.45)	звезда (0.57)
другой (0.50)	история (0.42)	система (0.56)
энергия (0.46)	общество (0.41)	другой (0.55)
время (0.43)	право (0.40)	много (0.49)
система (0.43)	мир (0.38)	время (0.43)
вселенная (0.40)	становиться (0.37)	учёный (0.40)
поле (0.39)	жизнь (0.36)	большой (0.40)
галактика (0.38)	политический (0.36)	год (0.39)

Таблица 3: Примеры плохих тем, полученных с помощью моделей с числом тем меньшим, чем в датасете. Темы состоят из нескольких интерпретируемых тем (часто связанных)

подтем более больших тем 6, часто темы были хорошими, но похожими.

Из эксперимента видно, что лучше обучать модель на большом числе тем. И при таком поиске хороших тем будет полезен банк тем: чтобы из большой модели выбрать хорошие темы, которые могут быть и чем-то похожи, но всё же отличаться (подтемы более большой темы).

5.2.3. Сравнение автоматических способов восстановления полного набора тем

Цель эксперимента — проверить, что с помощью оценок качества тем можно автоматически отбирать из моделей хорошие темы, или, по крайней мере, упрощать работу человека при анализе путём отсеивания части плохих тем сохранением хороших.

Обучились 10 моделей по 100 тем с разной начальной инициализацией. Далее глазами были просмотрены списки топ слов тем всех моделей: были определены интерпретируемые темы,

Биология	Социология	Астрономия
клетка (0.88)	человек (0.67)	звезда (0.68)
человек (0.57)	социальный (0.56)	земля (0.55)
другой (0.54)	общество (0.44)	год (0.51)
ген (0.44)	другой (0.44)	другой (0.51)
система (0.42)	страна (0.43)	галактика (0.48)
новый (0.40)	политический (0.43)	планета (0.46)
организм (0.39)	становиться (0.38)	время (0.45)
получать (0.37)	новый (0.37)	вселенная (0.45)
год (0.36)	право (0.37)	большой (0.41)
вид (0.34)	мир (0.33)	много (0.36)

Таблица 4: Примеры плохих тем, полученных с помощью моделей с числом тем меньшим, чем в датасете. Темы интерпретируемые, но в топ-словах есть слова, которые на самом деле не относятся к теме (это либо фоновые слова, либо слова из близких тем)

Биология	Физика	Земля
ген (1.79)	частица (2.35)	земля (1.07)
клетка (1.74)	кварк (1.39)	вода (1.04)
организм (1.04)	модель (1.22)	микроорганизм (0.79)
белок (0.93)	бозон (0.96)	газ (0.69)
человек (0.82)	нейтрино (0.96)	происходить (0.52)
бактерия (0.67)	масса (0.96)	жизнь (0.52)
другой (0.61)	стандартный (0.91)	атмосфера (0.52)
система (0.61)	взаимодействие (0.82)	год (0.51)
мутация (0.54)	эксперимент (0.71)	кислород (0.51)
болезнь (0.54)	физика (0.69)	океан (0.50)

Таблица 5: Примеры идеально восстановленных тем с помощью моделей с числом тем большим, чем в датасете (все топ-слова из первых десяти относятся к теме)

Культура (1)	Культура (2)	Культура (3)
культура (0.86)	культура (2.46)	фильм (2.82)
фольклор (0.61)	город (1.95)	кино (1.54)
другой (0.46)	пространство (1.39)	кинематограф (0.73)
праздник (0.43)	культурный (0.89)	культура (0.65)
событие (0.43)	городской (0.81)	массовый (0.61)
становиться (0.43)	новый (0.79)	зритель (0.53)
человек (0.42)	современный (0.58)	становиться (0.47)
игрушка (0.39)	другой (0.52)	книга (0.44)
культурный (0.36)	форма (0.48)	культовый (0.40)
мир (0.35)	общество (0.48)	тема (0.38)

Физика (1)	Физика (2)	Физика (3)
частица (3.27)	теория (3.22)	частица (2.67)
поле (2.27)	пространство (0.92)	кварк (1.23)
магнитный (1.57)	закон (0.84)	симметрия (1.18)
энергия (1.51)	физика (0.75)	пространство (0.77)
электрон (1.12)	математический (0.72)	свойство (0.74)
взаимодействие (0.99)	уравнение (0.66)	нейтрино (0.73)
симметрия (0.88)	время (0.66)	теория (0.72)
физика (0.82)	система (0.66)	три (0.69)
атом (0.78)	вселенная (0.64)	два (0.66)
элементарный (0.64)	эйнштейн (0.64)	взаимодействие (0.66)

История (1)	История (2)	История (3)
война (2.10)	война (2.34)	церковь (0.98)
советский (0.92)	германия (1.04)	король (0.64)
сталин (0.84)	первый (0.70)	власть (0.60)
год (0.61)	советский (0.69)	святой (0.56)
первый (0.59)	мировой (0.58)	папа (0.53)
власть (0.49)	немецкий (0.57)	время (0.51)
страна (0.48)	партия (0.56)	век (0.50)
революция (0.48)	история (0.56)	новый (0.40)
политический (0.43)	становиться (0.53)	католический (0.39)
мировой (0.43)	политический (0.53)	человек (0.37)

Таблица 6: Похожие темы/подтемы более большой темы, которые удалось найти с помощью моделей с большим, чем в датасете, числом тем

Quality function	# topics, %*	# good / # bad	# lost good, %**
Purity	21.5	7.75	31.2
Intratext	34.0	4.24	23.1
Newman	20.1	4.30	48.6
Purity	71.9	**2.79**	1.2
Intratext	69.4	**2.67**	2.3
Newman	69.0	**1.90**	1.7
Human	100.0	1.35	0

Таблица 7: Насколько функции качества тем способны повторять человеческие оценки при отборе хороших тем. С помощью оценок качества для заведомо хороших тем датасета (эталонные темы) подбирался порог, по которому происходил отбор тем. Результаты усреднялись по 10 обученным моделям с разной начальной инициализацией. В таблице не приведена функция Contrast, потому что результаты, полученные с её помощью, не шли в сравнение с приведёнными результатами для Purity и когерентностей. Те метрики, которые приведены в таблице, оценивают качество *тем*, а не целых моделей. Поэтому такая популярная мера качества тематических моделей, как перплексия, в таблице не приведена. В первом блоке строчек — лучшие результаты по соотношению хороших тем к плохим среди отобранных, когда разрешалась потеря хороших тем. Во втором блоке — лучшие результаты при условии, что не допускается потеря хороших тем. В последней строчке — способ отбора с помощью человека. Можно видеть, что даже при условии сохранения всех хороших тем (без учёта дубликатов) методам автоматической оценки *удалось повысить отношение хороших тем по отношению к плохим*.Лучший результат показала функция Purity, но стоит отметить, что предпочтительнее всё же использовать когерентности, так как они ближе к человеческому пониманию качества темы: методы когерентности принимают во внимание распределение слов темы по тексту. *В одной модели изначально было 100 тем, **в среднем по моделям было 17.3 уникальных хороших тем.

каждой такой теме поставлена в соответствие ближайшая к ней по смыслу из эталонных двадцати тем (иногда новая тема была просто похожа на одну из эталонных, иногда была подтемой эталонной темы). Таким образом, с помощью человека была получена разметка тем моделей на хорошие и плохие. Для тем моделей в то же время были посчитаны оценки качества: когерентности, чистота, контраст. По значениям качества на *эталонных темах* вычислялся порог, и все темы модели, у которых качество больше порога, считались хорошими, остальные — плохими. Полученная таким образом разметка сравнивалась с проведённой ранее вручную: сколько в результате автоматического отбора осталось хороших тем, сколько плохих, не были ли потеряны хорошие темы. Результаты приведены в таблице 7. Стоит отметить, как подбирались пороги для оценок качества: считалась персентиль по оценкам для эталонных тем (5, 10, 25 или 50) и домножалась на числовой коэффициент (0.1, 0.25, 0.50, 0.75, 0.8, 1.0, 1.2 или 1.5). Из такой комбинации выбиралась лучшая по тому, как автоматическая разметка соотносилась с человеческой.

5.2.4. Зависимость качества тематической модели от её инициализации

Цель эксперимента — на основе сравнения нескольких способов инициализации модели понять, есть ли польза от инициализации модели подмножеством заранее отобранных хороших тем.

Для обучения тематической модели необходимо задать начальные значения в матрице Ф. Можно же инициализировать матрицу Ф темами, которые заведомо хорошие. При этом хорошие темы могут быть получены из того же датасета в процессе исследования (например, найдены с помощью разных моделей и сохранены в банк хороших тем) или же хорошие темы могут быть взяты при работе с другой коллекцией документов.

Были зафиксированы гиперпараметры методов инициализации Arora и CDC (у CDC подобран так, чтобы у модели было от 50 до 100 тем, получилось возможным сделать 83 темы). Проинициализированы, обучены модели, посчитаны оценки качества.

Случайная модель обучалась 10 раз, с последующим усреднением результатов.

При обучении моделей с часть тем из банка происходило следующее: 3 раза среди 19 эталонных тем выбиралось нужное число тем для использования при инициализации, затем ещё 3 раза обучалась модель с выбранными темами (при разной инициализации оставшейся части

Method	Purity * 10	Intratext * 100	Newman
CDC	6.74	**1.96**	1.06
Arora	7.63	**1.18**	1.96
Банк тем (5 %)	0.12	0.16	4.14
Банк тем (10 %)	0.15	0.15	4.09
Банк тем (20 %)	0.24	0.15	4.12
Банк тем (80 %)	2.92	0.19	3.82
Банк тем (100 %)	3.76	0.22	3.49
Другой банк тем (5 %)	0.15	0.15	4.12
Другой банк тем (10 %)	0.25	0.16	4.17
Другой банк тем (20 %)	1.14	0.17	4.06
Другой банк тем (80 %)	3.50	0.19	3.98
Другой банк тем (100 %)	4.47	0.23	3.63
Случайная	0.09	0.16	4.16
Идеальная	3.77	0.21	3.91

Таблица 8: Иллюстрация качества методов инициализации модели. Для сравнения представлены случайный и идеальный способы инициализации (идеальный — инициализация модели априори известными темами датасета). Модели строились на 20 темах. Помощь банка при инициализации заключалась в том, что часть от 20 тем модели (часть указана в скобках в таблице) бралась из банка, остальные были случайными. Везде, где при инициализации была случайность, результаты усреднялись по нескольким моделям. Между делом можно заметить, что когерентность по топ словам плохо коррелирует с качеством моделей: например, у случайной показатель выше, чем у идеальной. По таблице видно, что CDC и Arora *с заметным отрывом опережают по качеству* (Purity и внутритекстовая когерентность) *способы инициализации при помощи банка*. Но видно, что чем больше тем брать из банка для инициализации, тем лучше качество модели (монотонная зависимость лучше прослеживается на примере Purity — внутритекстовая когерентность возрастает, но в случае родного банка не монотонно). При использовании родного банка повышения внутритекстовой когерентности удалось добиться только при 80 % тем, позаимствованных из банка. Хотя оценка Purity стала выше уже при 5 % тем. При использовании стороннего банка (Википедия) заметного по сравнению со случайной моделью улучшения качества удалось добиться при 20 % тем, взятых из банка.

матрицы). Темы брались как и из множества эталонных тем, так и из стороннего банка тем — полученных из документов другой коллекции (Википедия).

Для сравнения была обучена модель, инициализированная априори известными темами датасета.

Результаты эксперимента представлены в таблице 8.

5.2.5. Поиск новых тем с помощью неполного набора тем

Тематическим моделям, как отмечалось ранее, свойственна неполнота. Но при обучении одной модели часть тем может быть найдена успешно. Возникает вопрос, можно ли, имея информацию о некоторой части тем датасета, найти оставшиеся? Цель эксперимента — проверить, можно ли, имея на руках неполный банк тем, найти оставшиеся темы. Идея в том, чтобы для данной модели каким-то образом найти *документы, которые модель не может удачно тематизировать*. Предполагается, что документ естественного языка, как способ передачи мыслей, чувств и идей от человека к человеку, должен содержать небольшое число главных тем — тем, которые встречаются в документе с высокой вероятностью. Имея такие документы, которые плохо тематизируются моделью, можно *на этих* документах построить начальное приближение модели с помощью какого-нибудь метода поиска начальной инициализации тематической модели (в эксперименте используется Arora). Далее можно построить и обучить новую тематическую модель, проведя инициализацию Φ так, чтобы в ней присутствовали темы исходной модели и те, которые удалось найти на плохо тематизированных документах с помощью алгоритма Arora. После этого можно проверить:

- Как теперь тематизированы документы, которые раньше модель не могла удачно обрабо-

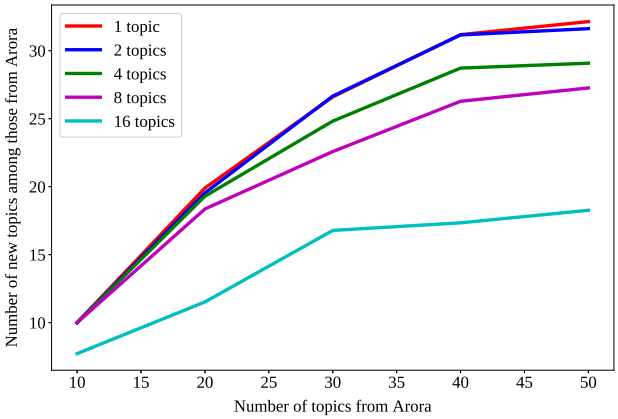


Рис. 3: Зависимость числа найденных новых хороших тем от числа хороших тем в исходной модели, которые получены с помощью алгоритма неслучайной инициализации Arora, при разных количествах априорно известных хороших тем в модели. Видно, что чем больше тем в модели изначально, тем меньше новых тем удаётся найти с помощью неслучайной инициализации Arora. Значит, с помощью Arora действительно можно находить хорошие темы.

тать.

- Есть ли такие документы, которые изначальной моделью тематизировались хорошо (то есть было небольшое число главных тем), но с добавленными от Arora темами стали тематизироваться плохо.

- Нашлись ли для добавленных тем документы, которые модель к ним относит. Иными словами, какие из добавленных от Arora тем оказались значимыми в плане исследования датасета. Ведь если к теме не относится ни один документ, то тему нельзя считать хорошей, а смысл заключается в поиске новых хороших тем.

Эксперимент по поиску новых тем состоит из нескольких частей. Первая заключается в том, чтобы выяснить, как зависит способность находить новые темы с помощью Arora от количества уже найденных хороших тем. Предполагается, что чем больше хороших тем содержит модель, тем меньше тем может быть найдено. Эксперимент проводился следующим образом. Назначалось число k, равное количеству хороших тем из банка, которые будут взяты для создания стартовой модели: 1, 2, 4, 8 или 16 тем (всего в датасете 19 тем). Далее 10 раз случайно из 19 тем банка выбиралось заданное число тем. Остальные $19 - k$ тем модели инициализировались случайно (для каждой подвыборки из k тем обучалось 5 моделей, и все дальнейшие результаты получены путём усреднения значений по этим моделям). Таким образом, модель на 19 темах, часть из которых эталонные, а остальные случайные — это исходная модель. Далее находились документы, которые такая модель не способна тематизировать. Как находились такие документы? Документ считался *плохо тематизированным, если среди первых трёх вероятностей в отсортированном по убыванию распределении* $p(t \mid d)$ *нет такой, которая бы была хотя бы в два раза больше следующей по порядку вероятности*. Логика за таким определением плохо тематизированных документов та же, что упоминалась ранее в общем описании эксперимента: допускается, чтобы у документа была одна главная тема, две или три (у которых вероятности значительно превосходят вероятности остальных тем для данного документа), но не больше. Так, документ, в котором темы распределены равномерно, будет плохо тематизированным и наоборот, документ, в котором затрагивается только одна тема, будет хорошо тематизированным. С помощью алгоритма Arora выполнялся поиск 50 тем (в не зависимости от количества плохо тематизированных документов). Результаты представлены в таблицах 9, 10. На графике 3 — наглядная иллюстрация данных из таблицы 10.

# Arora topics	# docs, %*	# new topics	# new topics, %	Purity	Intratext * 1000	Newman
0	15.8	0.0	-	0.77	1.74	4.47
1	15.8	1.0	100.0	0.76	1.72	4.49
5	16.2	5.0	100.0	0.72	1.96	4.02
10	17.4	10.0	100.0	0.68	2.28	3.90
20	16.2	19.54	97.7	0.43	4.64	2.19
30	16.8	26.66	88.9	0.37	7.16	1.79
40	15.5	31.16	77.9	0.38	8.37	1.54
50	14.6	31.62	63.2	0.41	9.64	1.43
-	21.8	-	-	0.59	9.20	1.87

Таблица 9: Поиск новых тем при 2 эталонных темах в исходной модели на 20 тем (10 % эталонных тем). В первом столбце указано количество тем, которые добавлялись к исходной модели от тем, найденных с помощью Arora. Таким образом, в первой строчке — результаты для модели без добавленных новых тем. Цель — чтобы увеличивалось количество хорошо тематизированных документов при добавлении новых тем и чтобы среди вновь добавленных к модели тем не оказывалось таких, к которым в итоге не относится ни один документ. В одном из столбцов указано процентное отношение тем, к которым модель отнесла ненулевое число документов, к общему числу добавленных к модели тем. Видно, что до какого-то момента увеличивается число правильно тематизированных документов, далее начинает снижаться. В то же время число новых хороших тем постепенно выходит на плато (а процентное соотношение снижается). Значит, спад числа правильно тематизированных документов может быть вызван тем, что среди добавляемых тем большая часть лишних, и они приводят к ухудшению модели. К сожалению, оценки качества тем Purity, внутритекстовая когерентность и когерентность по топ словам не позволяют определить момент, когда надо перестать добавлять темы, найденные Arora: Purity монотонно спадает, внутритекстовая когерентность увеличивается, Newman тоже монотонно спадает. Монотонное увеличение внутритекстовой когерентности может быть вызвано тем, что в модели много неплохих, но похожих тем: способ оценки качества относится к теме и не принимает во внимание факт, что в модели много похожих тем. В последней строчке для сравнения приведены число хорошо тематизированных документов и показатели качества для модели с темами только от Arora (то есть модель на 50 темах, с инициализацией по Arora). Видно, что число хорошо тематизированных документов только в случае Arora больше, но в данном эксперименте главное не это. Главное — принципиальная возможность нахождения новых тем, о чём свидетельствуют увеличивающиеся показатели числа хорошо тематизированных документов. *15.8 % документов хорошо тематизировались исходной моделью (всего в датасете 3446 документов).

# Arora topics	# topics in model		
	1	2	4
0	0.0	0.0	0.0
1	1.0	1.0	1.0
5	5.0	5.0	5.0
10	10.0	10.0	10.0
20	19.9	19.54	19.3
30	26.6	26.66	24.82
40	31.18	31.16	28.72
50	32.14	31.62	29.08

Таблица 10: Зависимость числа найденных новых хороших тем от числа хороших тем в исходной модели. *Модели по Arora строились на 50 тем.

Method	1^{st} strategy	2^{d} strategy	3^{d} strategy
Purity	Sp(−10)	Sp(−1)Sm(1)Dc(10^5)	Sp(−2)
Perplexity	Sm(0.01)	Dc(100)	Dc(10)
Bank	Sm(0.1)Sp(−1)Dc(10^5) ObO	Sm(0.1)Sp(−0.1)Dc(10^5)	Sm(0.1)Sp(−0.1)Dc(10^3)

Таблица 11: Первые три лучших стратегии обучения, найденные по оценка качества моделей с помощью способа, указанного в первом столбце (датасет «ПостНауки»).

5.2.6. Валидирование моделей с помощью банка тем (стратегии обучения)

На исследование одного датасета, на поиск и отбор всех хороших тем в нём может уходить много времени. Но можно ли как-то, зная набор хороших тем для одного датасета (которые не важно как были отобраны: либо с помощью тематических моделей, либо с помощью людей), найти все темы, затрагиваемые в другом датасете? Кажется, что в случае, когда датасеты *принципиально* не отличаются друг от друга (то есть стиль текста одинаков, по порядку одинаковы длины документов), такое априорное знание, пусть и о темах другой текстовой коллекции, может помогать. Цель этого эксперимента — выяснить, можно ли с помощью банка тем на одном датасете подбирать оптимальную стратегию обучения для моделей на другом датасете.

Эксперимент происходил на датасетах «ПостНауки» и Википедии 5.2.1. С помощью APTM подхода определяются несколько стратегий обучения как последовательности регуляризаторов: модель суммарно обучается несколько итераций, на первых итерациях работает один регуляризатор, на последующих нескольких итерациях работает другой регуляризатор (с или без первого), и так до конца обучения. Множество всех возможных стратегий ограничивалось единичными регуляризаторами и упорядоченными композициями регуляризаторов из множества {Декоррелирование тем (обозначение Dc от Decorrelation), Разреживание предметных тем (обозначение Sp от Sparse) Ф, Сглаживание фоновых тем Ф (обозначение Sm от Smooth)}, таких что в композиции участвуют все регуляризаторы из множества и регуляризаторы в композиции не повторяются. Таким образом, максимальное число регуляризаторов в одной композиции не превышало трёх. При этом в случае нескольких регуляризаторов в композиции у неё есть ещё то свойство, что допускается либо последовательное включение-выключение регуляризаторов (то есть на каждой итерации обучения работает только один регуляризатор) без повторения либо одновременная работа всех регуляризаторов на протяжении всего процесса обучения модели. Сколько было стратегий? При использовании стратегий только с одним регуляризатором у регуляризатора декоррелирования варьировался параметр на множестве из 6 значений, у регуляризаторов сглаживания и разреживания тоже варьировались параметры, у каждого по одному на множестве из 5 значений. При трёх регуляризаторах в композиции у каждого множество значений для варьирования параметра ограничивалось тремя значениями (для ускорения эксперимента). Также в качестве стратегии обучения рассматривалась и модель без регуляризаторов. Таким образом, получаем $2 * \left(1 + 6 + 5 + 5 + 6 * (3 * 3 * 3)\right) = 2 * 179 = 358$.

Цель эксперимента, как говорилось выше, состоит в том, чтобы проверить, можно ли находить лучшую стратегию обучения с помощью банка тем. Значит, надо уметь оценивать качество модели, полученной в результате обучения по стратегии. За меру оценки качества с помощью банка была принята такая: медиана среди минимальных расстояний до эталонных хороших тем от тем модели. Интуиция за этим состоит в том, что чем меньше расстояния от тем модели до хороших тем, тем лучше модель (конечно, при условии, что хороших тем собрано достаточное количество, пусть, возможно, и не все). Также для сравнения при оценке качества моделей использовались медианное значение Purity среди тем модели и перплексия модели. Результаты по сравнению стратегий обучения с помощью банка представлены в таблицах 11, 12 и 13.

5.2.7. Ускорение процесса поиска полного набора тем

Главный вопрос для исследования в этом эксперименте — выяснить, возможно ускорить процесс воссоздания полного набора тем с помощью банка тем.

Изначально банк тем брался пустой. И начиналось обучение моделей на описанном датасете. Начиналось с инициализации модели (либо все темы инициализировались случайно, либо половина тем были взяты как темы, найденный с помощью Arora или CDC). Каждая модель обуча-

Method	1^{st} strategy	2^{d} strategy	3^{d} strategy
Purity	Sp(−10)	Sp(−2)	Dc(10^5)Sp(−1)Sm(1)
Perplexity	Sm(1)Dc(10)Sp(−1)	Sm(1)Dc(10^3)Sp(−1)	Dc(10)Sm(1)Sp(−1)
Bank	Sm(10)	Dc(10^5)Sm(1)Sp(−1)	Dc(10^5)Sp(−1)Sm(1)

Таблица 12: Первые три лучших стратегии обучения, найденные по оценка качества моделей с помощью способа, указанного в первом столбце (датасет Википедии).

Method	Correlation	P-value (H_0: uncorrelated)
Purity	0.763	$2.49 * 10^{-69}$
Perplexity	0.296	$1.10 * 10^{-8}$
Bank	−0.148	$5.13 * 10^{-3}$

Таблица 13: Спирмановские корреляции между значениями функций качества на моделях при обучении по описанному набору стратегий для датасета «ПостНауки» и Википедии.

ется в течение 20 итераций. В моделях по 100 тем. В конце обучения оценивается правдоподобие модели и другие упомянутые оценки качества. Далее среди тем обученной модели выбираются те, *качество которых по внутритекстовой когерентности не менее* 90 *персентили среди качеств всех тем модели*. Эти темы считаются хорошими. Среди этих отобранных тем ищутся те, которые не похожи на темы, которые хранятся в банке. Расстояние между темами оценивалось по метрике Жаккара, при этом учитывались только те слова, вероятность которых была больше равномерной $p(w \mid t) > \frac{1}{W}$. Порог устанавливался равным 0.5: *если расстояние между темой модели и ближайшей темой банка не меньше порога, то тема модели считается новой*. Все новые хорошие темы добавляются в банк.

Таким образом был устроен процесс отбора хороших тем с целью создания полного набора тем.

Результаты.

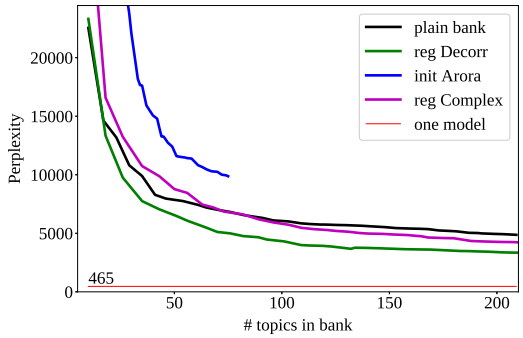
Результаты эксперимента представлены на графиках 4a, 4b, 4c, 7a, 7e, 7f, 7b.

Поясним обозначения на графиках 4a, 4b. Имя *plain bank* означает набор тем, которые хранятся в банке при отсутствии регуляризации и при начальной инициализации моделей; *one model* означает, что график показывает среднее значение перплексии для обучаемых без регуляризации и специальной инициализации моделей; обозначение *reg Decorr* отражает состояние банка при использовании регуляризатора докоррелирования тем для обучения моделей; *reg Complex* значит, что банк создавался при действии двух регуляризаторов: декорреляции и сглаживания тем; *init Arora* означает, что, вместо случайной инициализации, использовалась инициализация по алгоритму Arora для половины из тем модели; *init CDC* — инициализация половины тем моделейс помощью алгоритма CDC. Глядя на графики, 4a, 4b можно заключить, что автоматическое определение качества тем вместе с использованием регуляризации может ускорить процесс поиска полного набора тем: определённая регуляризация позволила получить лучшее правдоподобие модели банка, в сравнении с банком, собранным без применения регуляризации. И само значение правдоподобия сходилось быстрее: понадобилось меньшее количество моделей для получения более хорошего результата.

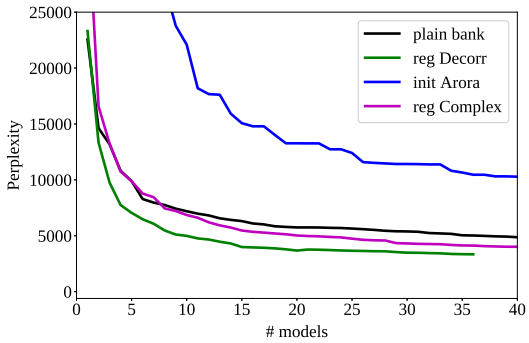
Глядя на графики 4b, 4c, можно заключить, что с помощью обучения нескольких моделей и с отбором хороших тем можно определить примерное число тем в коллекции. Когда стабилизируется перплексия 4b, это означает, что больше новые темы не находятся, и процесс обучения моделей можно остановить в данной точке. По номеру моделей в точке остановки улучшения перплексии на графике 4b можно найти число тем на графике 4c: это примерное число тем в полном наборе тем. Таким образом, число тем в наборе оказалось в отрезке [50, 150]. Оценка не точная, но по порядку величины совпадает с числом тем в датасете 19.

Графики 7a, 7c, 7d показывают, что темы в полных наборах на самом деле имеют лучшие показатели когерентностей по топ-словам и внутритекстовой в сравнении с темами обычных тренируемых моделей.

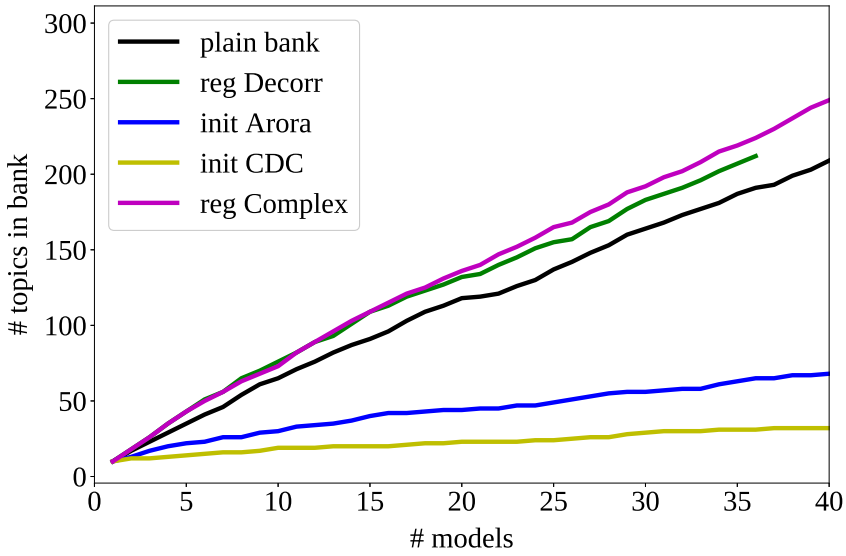
Графики 5a, 5b, 5c, 5d отражают отличия процессов сбора банков в зависимости от порога внутритекстовой когерентности, по которому темы добавляются в банк.



(a) Перплексия модели банка в зависимости от количества тем, добавленных в банк. График с инициализацией по CDC не показан, потому что перплексия такого банка оказалась заметно выше, чем представленные значения перплексии.

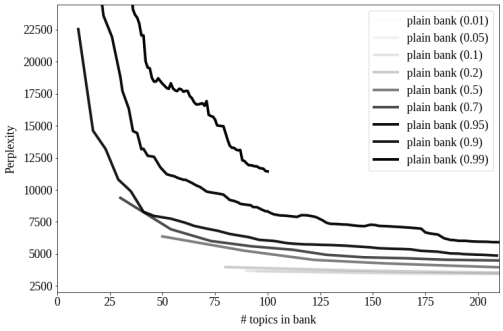


(b) Перплексия модели банка в зависимости от количества тренируемых моделей, которые использовались для создания банка. Регуляризация может помогать сходиться быстрее.

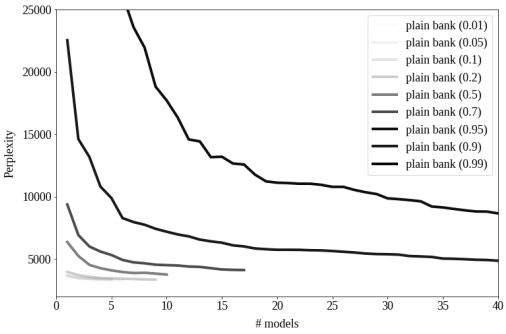


(c) Количество тем в банке в зависимости от количества тренируемых моделей. Количество тем линейно увеличивается, но перплексия на соседнем графике выше выходит на плато, что значит, что, хотя банк и растёт, новых тем в него больше не добавляется.

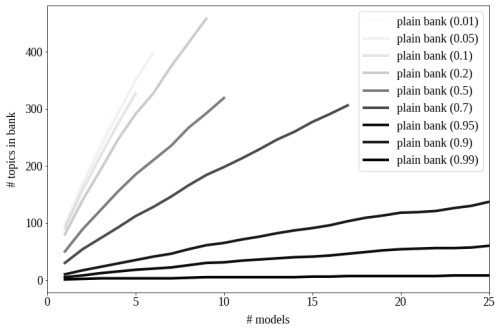
Рис. 4: Создание банка тем с помощью обучения нескольких моделей. Хорошие темы выбираются из каждой модели и добавляются в банк. Таким образом, банк постепенно растёт.



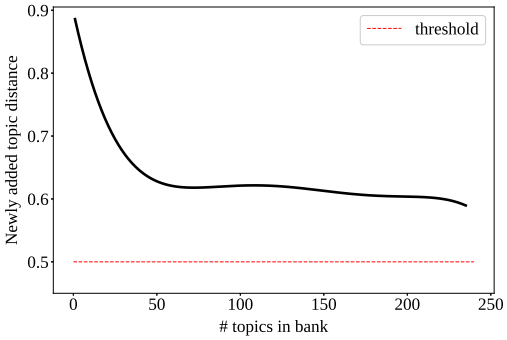
(a) Перплексия в зависимости от количества тем в банке для разных значений порога, по которому отбирались темы. Чем больше порог, тем более качественные темы должны отбираться, но перплексия итоговой модели получается выше. Это может свидетельствовать о том, что построенный процесс обучения не позволяет находить много различных хороших тем.



(b) Перплексия в зависимости от количества тренируемых моделей в банке для разных значений порога, по которому отбирались темы. Если проводить жёсткий отбор, то приходится тренировать больше моделей для сбора банка.



(c) Количество тем в банке в зависимости от числа обучаемых моделей для разных порогов при отборе тем. Чем жёстче отбор, тем медленнее добавляются темы, но во всех случаях банк линейно растёт. Это не значит, что добавляются новые темы. Начиная с какого-то момента добавляемые темы становятся линейно зависимыми от тем, уже добавленных в банк. На линейную зависимость при добавлении в банк темы не проверялись.



(d) Расстояние от вновь добавляемой в банк темы до ближайшей темы банка в зависимости от количества тем в банке. Пунктирная линия обозначает порог, по которому оценивалась близость между темами: только темы с расстоянием до ближайшей больше порога рассматривались как новые и могли быть добавлены в банк тем. Расстояние монотонно уменьшается, но порога не достигает, а выходит на плато. Возможно, стабилизация может свидетельствовать о том, что в банк больше не добавляются новые темы: при создании банка использовалась только модель одного вида, поэтому в результате обучения темы таких моделей отличались примерно на одну и ту же величину.

Рис. 5: Постепенное увеличение банка, накопление хороших тем.

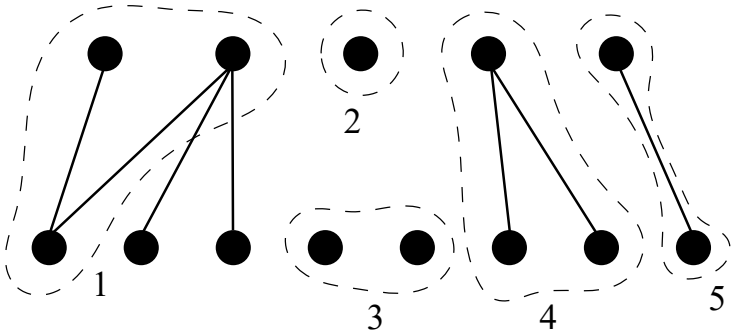


Рис. 6: Отличия между иерархическим тематическим моделированием и банком тем. Верхний уровень точек представляет родительские темы (темы банка тем), нижний уровень точек — дочерние темы (темы вновь обученной модели). Ситуация 1 изображает соединение нескольких родительских тем. Это нормально для иерархии, но не для банка тем: исходные темы остаются в банке, новые темы не добавляются. Ситуация 2, это когда у темы нет дочерних тем. Это допустимо как для иерархии, так и для банка тем. Случай 3 описывает ситуацию, когда у дочерних тем нет родителей. Это неприемлемо для иерархии, но допустимо для банка тем: новые темы сразу же добавляются в банк. Расположение 4 — когда у одной темы несколько детей, а у этих детей единственный родитель (то есть одна тема расщепляется на несколько). Для иерархий это — обычная ситуация. Для банка же тем: родительская тема может быть заменена дочерними темами, если все они интерпретируемы. Ситуация 5 описывает случай, когда тема переходи на следующий уровень без изменений. Это допустимо для иерархий. Также и для банка тем: в банке остаётся лучшая тема из двух.

5.2.8. Валидирование моделей с помощью банка тем (ряд моделей)

Главная цель эксперимента следующая: имея ряд тематических моделей, ряд датасетов выяснить, способен ли банк тем определить такую тему, которая бы на всех датасетах показывала бы лучшее качество, чем другие темы. Для каждого из датасетов, таким образом, надо сделать две вещи: создать банк тем и провести оценку качества моделей с его помощью.

Создание банка тем.

Множественное обучение моделей — главная составляющая создания банка тем. Общее число моделей для обучения было выбрано заранее и положено равным 20. Когда тренируется новая модель, для всех её тем вычисляется когерентность. И 10 процентов тем с самым высоким значением внутритекстовой когерентности [19] извлекаются из модели. Эти темы считаются интерпретируемыми. Однако, так как оценка качества тем с помощью людей не проводилась (и не планировалась проводиться в данном эксперименте), *некоторые темы на самом деле могут и не быть интерпретируемыми*. Это гипотетическое предположение, что *каждая модель способна находить интерпретируемые темы* (а интерпретируемость может быть оценена с помощью когерентности). Итак, теперь у нас есть 10 тем, но не все эти темы будут добавлены в банк.

Далее, мы оцениваем связь между темами в банке и новыми темами. Пусть t обозначает некоторую тему из банка тем, и s — некоторую новую тему. Тогда $p(s \mid t)$ есть оценка того, что тема s является ребёнком темы t. Эти оценки принадлежности дочерних тем родительскими получаются с помощью иерархического тематического моделирования (можно даже заметить сходство между следующим уравнением и уравнением, описывающим иерархическую модель 1):

$$\underbrace{p(w \mid t)}_{\phi^{bank}_{wt}} = \sum_{s \in S} \underbrace{p(w \mid s)}_{\phi^{new}_{ws}} \underbrace{p(s \mid t)}_{\psi_{st}} \tag{2}$$

Если $\psi_{st} > 1/|S|$, то тема t предполагается родителем темы s. Иначе — нет. Есть несколько возможных случаев

- у темы s есть более одного родителя. В этом случае тема s не будет добавлена в банк, так как она является линейной комбинацией нескольких тем банка.
- у темы s нет родителей. В этом случае тема может быть добавлена в банк.

- у темы s есть только один родитель. В этом случае родительская тема может быть заменена на новую тему, если качество новой темы (внутритекстовая когерентность) выше, чем у родительской.

Среди 10 тем с самым высоким показателем когерентности только те, которые удовлетворяют описанному выше условию связи (не имеют родителей или имею только одного родителя) оставляются на рассмотрении. Темы без родителей сразу добавляются в банк, темы же с только одним родителем могут заменить соответствующую родительскую темы в случае, если её когерентность выше, чем у родителя.

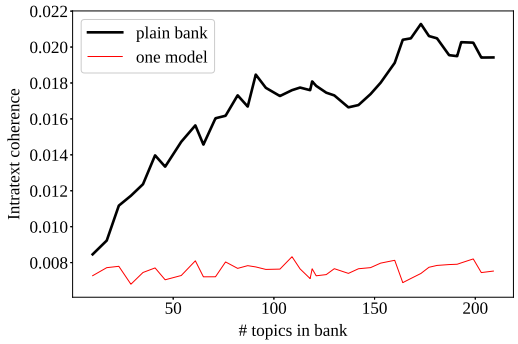
На графике 7 представлена визуализация, показывающая, что в банке тем действительно собираются темы с более высоким значением когерентности по сравнению с темами тренируемых моделей. И не только внутритекстовая когерентность у тем банка выше (что очевидно, так как отбор тем базируется на значениях внутритекстовой когерентности), но и когерентность по топ-словам.

Валидация моделей.

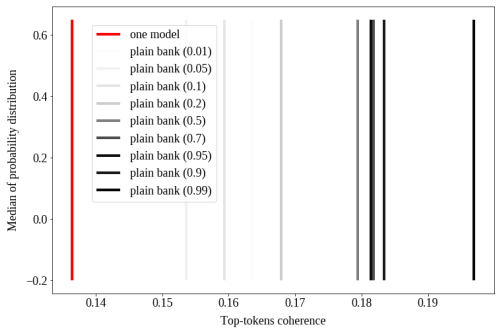
Банк тем нужен для оценки качества моделей. Список моделей для валидации представлен ниже.

- *plsa*: модель PLSA,
- *lda*: модель LDA с симметричными априорными распределениями,
- *sparse*: модель с регуляризатором разреживания[10] [10],
- *decorr*: модель с регуляризатором декоррелирования[10] [10],
- *bcg*: модель с двумя фоновыми темами и регуляризатором сглаживания для этих фоновых тем[10] [10],
- *regul1*: модель с несколькими регуляризаторами (декорреляция, разреживание, сглаживание)[10],
- *regul2*: модель с несколькими регуляризаторами (декорреляция, разреживание, сглаживание) и фиксированными относительными весами для всех датасетов (относительные веса потом пересчитываются в абсолютные, которые уже свои для каждого датасета, и именно абсолютные веса используются в алгоритмах получения численных решений для Φ и Θ),
- *arora*: модель с инициализацией Φ матрицы по алгоритму Arora [23],
- *cdc*: модель с инициализацией Φ матрицы по алгоритму CDC [26],
- *sel_aooa*: модель с декоррелирующим регуляризатором, затем с регуляризатором отбора тем [42], затем с регуляризатором разреживания. В моделях, которые описываются ниже, используются те же самые регуляризаторы. Чем они отличаются, так это способом совместного использования регуляризаторов, способом их комбинирования: в статье [42] было показано, что порядок регуляризаторов может оказывать весомое влияние на тематическую модель,
- *sel_cp*: веса регуляризаторов декоррелирования и отбора тем увеличиваются с итерацией EM алгоритма,
- *sel_ao*: регуляризаторы декоррелирования и отбора тем чередуются (то есть на каждой итерации обучения задействован лишь один из этих двух регуляризаторов),
- *sel_aocp*: регуляризаторы декоррелирования и отбора тем чередуются, и их веса τ при этом постепенно увеличиваются.

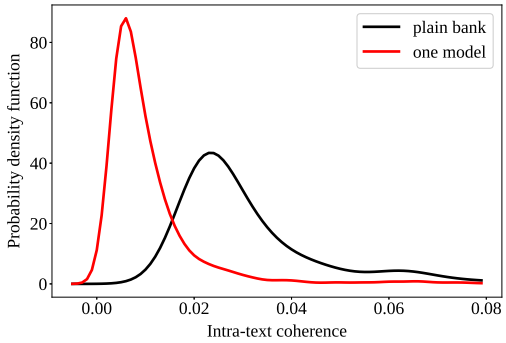
[10]Веса регуляризаторов были найдены по сетке из нескольких значений с помощью тренировки моделей с лишь одним регуляризатором.



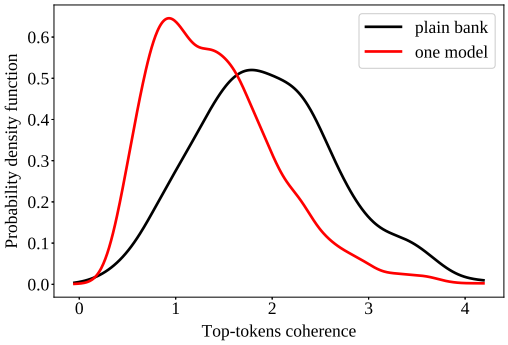
(a) Внутритекстовая когерентность в зависимости от количества тем в банке. Полный набор тем демонстрирует большие значения когерентности, чем обычные модели, которые использовались для создания банка.



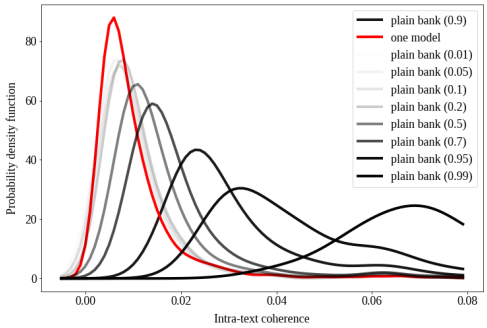
(b) Медиана распределения значений когерентности по топ словам для разных значений порога внутритекстовой когерентности, по которому в банк отбирались темы. Видно, что когерентность по топ словам растёт.



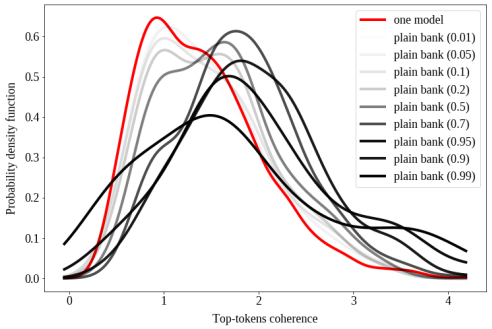
(c) KDE оценка функции вероятности по значениям внутритекстовой когерентности.



(d) KDE оценка функции вероятности по значениям когерентности по том словам.



(e) KDE оценка функции вероятности по значениям внутритекстовой когерентности.



(f) KDE оценка функции вероятности по значениям когерентности по топ словам.

Рис. 7: Четыре нижних графика показывают KDE оценки для внутритекстовой когерентности и когерентности по топ словам для тем банка и тем обычных моделей, которые использовались для построения банка (на примере датасета ПостНаука). Темы добавлялись в банк в зависимости от того, какое у них было значение внутритекстовой когерентности (добавлялись с более высоким значением). Поэтому внутритекстовая когерентность у банка ожидаемо выше. Однако по когерентности по том словам банк тоже превосходит обычные модели. Первые два графика из описываемых четырёх — для случая, описанного в тексте: когда у вновь обученной модели отбираются 10 процентов тем с самым высоким значением когерентности. Последние два содержат много линий для банка тем: они изображают случаи, когда из моделей извлекалось разное число тем для добавления в банк: от 99 процентов тем модели до 1 процента.

Больше информации о регуляризаторах можно найти в [10], а также в секции 3.1, посвящённой регуляризаторам.

Способы оценки качества моделей с использованием банка тем.

Обозначим за B темы в банке тем, за T — темы текущей тематической модели. В работе предлагаются несколько функций качества моделей, основанных на банке тем.

$$\begin{aligned} \text{recall@bank} &= \frac{\#\{t \in B : t \in T\}}{|B|} \\ \text{coherence@bank} &= \frac{\#\{t \in T : t \in B\}}{|T|} \\ \text{precision@bank} &= \frac{\#\{t \in B : t \in T\}}{|T|} \end{aligned} \tag{3}$$

То есть recall@bank означает, как много тем банка модель смогла найти, coherence@bank означает, как много тем модели также были в банке тем (и потому считаются интерпретируемыми), precision@bank означает долю, которую составляют темы модели, находящиеся также и в банке тем, от всех тем модели.

На данном этапе ещё не до конца понятно, как считать recall@bank, coherence@bank и precision@bank: надо задать расстояние между темами $\rho(t_1, t_2)$. Будем оценивать расстояние по мере Жаккара.

$$\begin{aligned} \rho(t_1, t_2) = 1 - \sum_{w \in \text{Ker}_{12}} \min_{i \in \{1,2\}} \big(p(w \mid t_i)\big) \Big/ \\ \left(\sum_{i=1}^{2} \sum_{w \in \text{Ker}_i \setminus \text{Ker}_{12}} p(w \mid t_i) + \sum_{w \in \text{Ker}_{12}} \max_{i \in \{1,2\}} \big(p(w \mid t_i)\big) \right) \end{aligned} \tag{4}$$

где $\text{Ker}_i \equiv \text{Ker}(t_i)$, $\text{Ker}_{12} \equiv \text{Ker}(t_1) \cap \text{Ker}(t_2)$ и $\text{Ker}(t)$ — это ядро темы t:

$$\text{Ker}(t) = \big\{w \in t : p(w \mid t) > 1/|W|\big\}$$

Будем считать две темы близкими, если расстояние между ними меньше некоторого порога: $h \in H \subseteq (0, 1]$, $|H| < \infty$. Очевидно, чем меньше порог, тем строже оценка близости. Определим отображение из порога h в неотрицательный вес, такое что меньшему порогу будет соответствовать больший вес:

$$w(h) = \begin{cases} 1/h^2, & h \le 0.8 \\ 0, & h > 0.8 \end{cases}$$

Рассмотрим метрику качества recall@bank (с другими рассуждения аналогичны). На данный момент мы имеем recall@bank как функцию от датасета d, модели m и порога h. Если мы проведём усреднение по порогам, то получим

$$\langle \text{recall@bank}(m, d, h) \rangle_h = \frac{\sum\limits_{\eta \in H} w(\eta) \cdot \text{recall@bank}(m, d, \eta)}{\sum\limits_{\eta \in H} w(\eta)}$$

И в итоге, усредняя ещё по датасетам, получаем финальную оценку качества, зависящую только от модели:

$$\langle \text{recall@bank}(m, d, h) \rangle_{d,h} = \frac{1}{|\mathscr{D}|} \sum_{\delta \in \mathscr{D}} \frac{\langle \text{recall@bank}(m, \delta, h) \rangle_h}{\sum\limits_{\mu \in \mathscr{M}} \langle \text{recall@bank}(\mu, \delta, h) \rangle_h} \tag{5}$$

Где $\mathscr{D}$ — это множество датасетов 2 (без датасета Википедии), и $\mathscr{M}$ — это заранее зафиксированное множество моделей. Иллюстрация 8 поясняет используемый способ усреднения по датасетам.

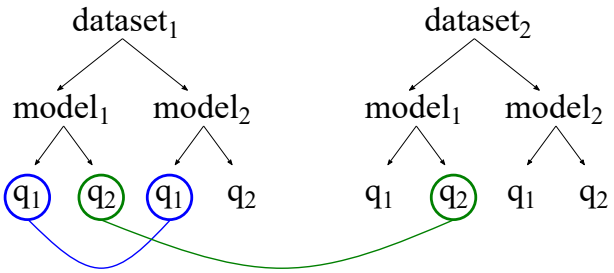


Рис. 8: Иллюстрация к выражению 5: есть много датасетов, для каждого из которых тренируется много моделей, для каждой из которых посчитаны оценки качества q_i. Синий цвет означает усреднение оценки качества по моделям, тренируемым на одном датасете. Зелёный цвет означает усреднение оценки качества одной модели, но тренируемой на разных датасетах.

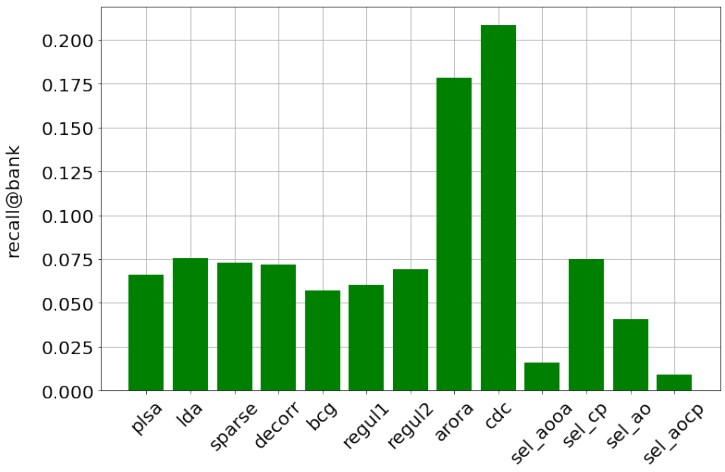


Рис. 9: Оценки качества моделей относительно банка тем, усреднённые по датасетам. Post_Science означает датасет ПостНауки, 20NG_natural_order означает датасет Twenty Newsgroups.

Итоговые результаты представлены на рисунке 9 (с только одной метрикой качества из 3, так между метриками не было существенного различия). Можно видеть, что модели *arora* с *cdc* превосходят остальные.

В отдельности для нескольких датасетов (то есть без усреднения по датасетам) оценки качества моделей представлены на рисунке 10.

По предположению, среди ряда моделей существовала та, которая лучше всего подстраивается под любые данные. Таким образом, банк тем помог среди ряда моделей найти такую модель.

С целью исключить человеческое влияние на эксперименты, все эксперименты проводились с *полностью автоматическим* способом создания множеств интерпретируемых тем. Однако, для практического применения банка тем рекомендуется проводить оценку качества тем с привлечением людей. Но и когерентность тем при этом остаётся полезной, так как она может упростить работу асессора.

Также стоит отметить, что в создание банка тем был вовлечён только один вид моделей (PLSA). Это было сделано, с одной стороны, для простоты: ранее в работе говорилось, что от банка тем не требуется того, чтобы он содержал в себе абсолютно все интерпретируемые темы датасета. Однако, для создания лучших, более качественных банков тем рекомендуется использовать разные тематические модели. Разные не только в способе обучения, но и по части гиперпараметров, например, по числу тем.

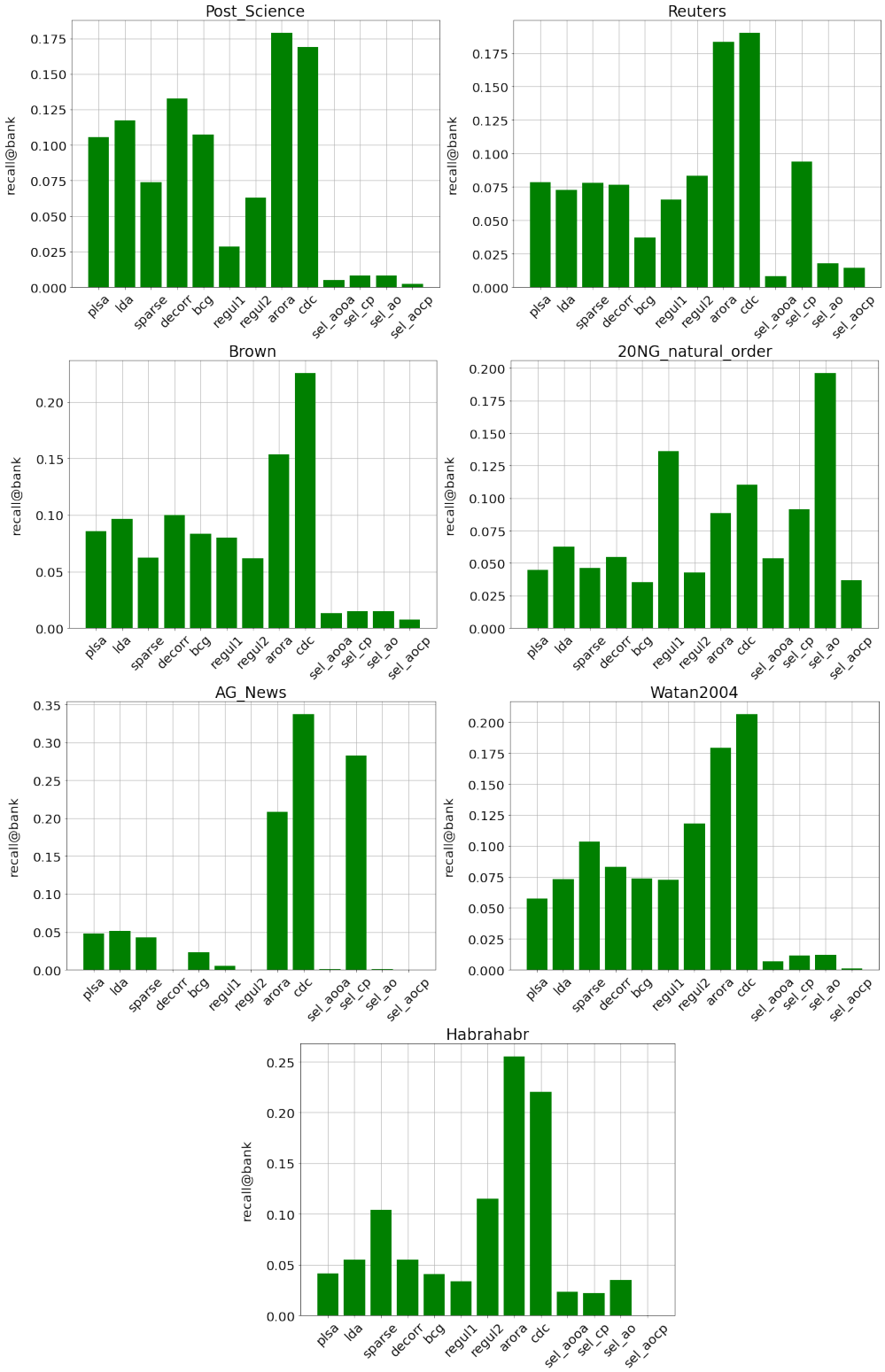


Рис. 10: Оценки качества моделей относительно банка тем, без усреднения по датасетам, для нескольких датасетов.

Заключение

Тематическое моделирование используется для исследования данных. Однако тематические модели неполны и неустойчивы. Много работ посвящено преодолению этих недостатков путём изменения способа тренировки модели. Так или иначе, обычно всё сводится к вариьрованию параметров модели в поисках лучшей. В данной же работе представлен более простой, более понятный и прямой путь исследования коллекции документов с помощью тематических моделей, который фокусируется не на тренировке, а на оценке качества моделей. Предлагаемый подход состоит из двух шагов. Сначала, полностью без учителя или с частичным привлечением учителя создаётся набор интерпретируемых тем с помощью множественного обучения моделей. Далее, собранные темы участвуют в автоматической оценке качества новых моделей.

В работе также произведено сравнение способов автоматической оценки качества тем для отбора интерпретируемых тем модели, исследованы некоторые возможности применения отобранных интерпретируемых тем для построения в конечном итоге идеальной модели для данной текстовой коллекции.

Среди других результатов работы:

- использование автоматических способов оценки интерпретируемости тем (например когерентности) позволяет отсеивать часть неинтерпретируемых тем модели, облегчая последующий анализ человеком,
- для начальной инициализации модели лучше применять техники CDC или Arora, чем часть из заранее отобранных хороших тем,
- имея на руках часть тем коллекции, можно использовать их для более эффективного поиска оставшихся тем,
- с помощью собранных тем на одном датасете можно подбирать лучшую стратегию обучения для моделей на другом датасете,
- банк тем — предлагаемый инструмент для тематического моделирования, позволяющий сохранять темы, которые находят модели,
- лучше обучать модели на заведомо большем числе тем, чем есть в датасете, это позволяет находить большее число тем, причём среди тем встречаются явно родительские и дочерние.

Остаётся открытым вопрос, как за один раз, с минимальным привлечением человека, построить такую тематическую модель, в которой все темы интерпретируемы, различны, и все темы которой — это именно все темы датасета, и больше найти новых тем нельзя. В работе описаны возможности, как можно с меньшими затратами идти к цели нахождения всех тем. Но готового рецепта обучения идеальной модели пока нет, и это тема дальнейших исследований. Можно ли определить число тем в коллекции документов с помощью банка тем? Как улучшить процесс сбора когерентных тем от множественного обучения моделей (сделать темы лучше, а весь процесс быстрее)? Какой эффект могут оказать регуляризация и неслучайная инициализация матрицы Ф тренируемых для создания банка моделей на итоговый банк тем? Эти вопросы тоже открыты и являются предметом будущих исследований.

Список литературы

[1] Blei D. M. Probabilistic topic models // Communications of the ACM. 2012. Vol. 55, no. 4. P. 77–84.

[2] Lautamatti L. Observations on the development of the topic in simplified discourse // AFinLAn vuosikirja. 1978. P. 71–104.

[3] Rubin T. N., Chambers A., Smyth P., Steyvers M. Statistical topic models for multi-label document classification // Machine learning. 2012. Vol. 88, no. 1-2. P. 157–208.

[4] Wang C., Blei D. M. Collaborative topic modeling for recommending scientific articles // Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. 2011. P. 448–456.

[5] Ianina A., Golitsyn L., Vorontsov K. Multi-objective topic modeling for exploratory search in tech news // Conference on Artificial Intelligence and Natural Language / Springer. 2017. P. 181–193.

[6] Varshney D., Kumar S., Gupta V. Modeling information diffusion in social networks using latent topic information // International Conference on Intelligent Computing / Springer. 2014. P. 137–148.

[7] Daud A., Li J., Zhou L., Muhammad F. Knowledge discovery through directed probabilistic topic models: a survey // Frontiers of computer science in China. 2010. Vol. 4, no. 2. P. 280–301.

[8] Hofmann T. Probabilistic latent semantic indexing // Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. 1999. P. 50–57.

[9] Blei D. M., Ng A. Y., Jordan M. I. Latent dirichlet allocation // Journal of machine Learning research. 2003. Vol. 3, no. Jan. P. 993–1022.

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

[11] Steyvers M., Griffiths T. Probabilistic topic models // Handbook of latent semantic analysis. 2007. Vol. 427, no. 7. P. 424–440.

[12] Mimno D., Wallach H. M., Talley E. et al. Optimizing semantic coherence in topic models // Proceedings of the conference on empirical methods in natural language processing / Association for Computational Linguistics. 2011. P. 262–272.

[13] De Waal A., Barnard E. Evaluating topic models with stability. 2008.

[14] Koltcov S., Koltsova O., Nikolenko S. Latent dirichlet allocation: stability and applications to studies of user-generated content // Proceedings of the 2014 ACM conference on Web science / ACM. 2014. P. 161–165.

[15] Greene D., O'Callaghan D., Cunningham P. How many topics? stability analysis for topic models // Joint European Conference on Machine Learning and Knowledge Discovery in Databases / Springer. 2014. P. 498–513.

[16] Ianina A., Vorontsov K. Regularized multimodal hierarchical topic model for document-by-document exploratory search // 2019 25th Conference of Open Innovations Association (FRUCT) / IEEE. 2019. P. 131–138.

[17] Balagopalan A. Improving topic reproducibility in topic models. University of California, Irvine, 2012.

[18] Koltcov S., Nikolenko S. I., Koltsova O. et al. Stable topic modeling with local density regularization // International Conference on Internet Science / Springer. 2016. P. 176–188.

[19] Alekseev V., Bulatov V., Vorontsov K. Intra-text coherence as a measure of topic models' interpretability // Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference Dialogue. 2018. P. 1–13.

[20] Mehta V., Caceres R. S., Carter K. M. Evaluating topic quality using model clustering // 2014 IEEE Symposium on Computational Intelligence and Data Mining (CIDM) / IEEE. 2014. P. 178–185.

[21] Agrawal A., Fu W., Menzies T. What is wrong with topic modeling? and how to fix it using search-based software engineering // Information and Software Technology. 2018. Vol. 98. P. 74–88.

[22] Cao J., Xia T., Li J. et al. A density-based method for adaptive LDA model selection // Neurocomputing. 2009. Vol. 72, no. 7-9. P. 1775–1781.

[23] Arora S., Ge R., Kannan R., Moitra A. Computing a nonnegative matrix factorization–provably // Proceedings of the forty-fourth annual ACM symposium on Theory of computing / ACM. 2012. P. 145–162.

[24] Arora S., Ge R., Moitra A. Learning topic models–going beyond SVD // 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science / IEEE. 2012. P. 1–10.

[25] Arora S., Ge R., Halpern Y. et al. A practical algorithm for topic modeling with provable guarantees // International Conference on Machine Learning. 2013. P. 280–288.

[26] Dobrynin V., Patterson D., Rooney N. Contextual document clustering // European Conference on Information Retrieval / Springer. 2004. P. 167–180.

[27] AlSumait L., Barbará D., Domeniconi C. On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking // 2008 eighth IEEE international conference on data mining / IEEE. 2008. P. 3–12.

[28] Bruggermann D., Hermey Y., Orth C. et al. Storyline detection and tracking using dynamic latent dirichlet allocation // Proceedings of the 2nd Workshop on Computing News Storylines (CNS 2016). 2016. P. 9–19.

[29] Keane N., Yee C., Zhou L. Using topic modeling and similarity thresholds to detect events // Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation. 2015. P. 34–42.

[30] Hoffman T. Probabilistic latent semantic indexing // Proceedings of the 22nd Annual ACM Conference on Research and Development in Information Retrieval, 1999. 1999. P. 50–57.

[31] Vorontsov K., Frei O., Apishev M. et al. Bigartm: Open source library for regularized multimodal topic modeling of large collections // International Conference on Analysis of Images, Social Networks and Texts / Springer. 2015. P. 370–381.

[32] Vorontsov K., Potapenko A. Tutorial on probabilistic topic modeling: Additive regularization for stochastic matrix factorization // International Conference on Analysis of Images, Social Networks and Texts / Springer. 2014. P. 29–46.

[33] Chirkova N., Vorontsov K. Additive regularization for hierarchical multimodal topic modeling // Journal Machine Learning and Data Analysis. 2016. Vol. 2, no. 2. P. 187–200.

[34] Koltcov S. Application of Rényi and Tsallis entropies to topic modeling optimization // Physica A: Statistical Mechanics and its Applications. 2018. Vol. 512. P. 1192–1204.

[35] Newman D., Lau J. H., Grieser K., Baldwin T. Automatic evaluation of topic coherence // Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics / Association for Computational Linguistics. 2010. P. 100–108.

[36] Lau J. H., Newman D., Baldwin T. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality // Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics. 2014. P. 530–539.

[37] Lewis D. D. Reuters-21578 [dataset]. 1997. URL: http://www.daviddlewis.com/resources/testcollections/reuters21578/.

[38] Kucera H., Francis W. N. Brown [dataset]. 1961. URL: http://korpus.uib.no/icame/brown/bcm.html.

[39] Lang K. 20 Newsgroups [dataset]. 1995. URL: http://qwone.com/~jason/20Newsgroups/.

[40] Gulli A. AG News [dataset]. URL: http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html.

[41] Abbas M. Watan-2004 [dataset]. URL: https://sites.google.com/site/mouradabbas9/corpora/text-corpora.

[42] Vorontsov K., Potapenko A., Plavin A. Additive regularization of topic models for topic selection and sparse factorization // International Symposium on Statistical Learning and Data Sciences / Springer. 2015. P. 193–202.