

УДК 004.8

DOI 10.21661/r-561564

Большаков Сергей Сергеевич

магистрант

Научный руководитель

Стуколов Сергей Владимирович

канд. физ.-мат. наук, доцент

ФГБОУ ВО «Кемеровский государственный университет»

г. Кемерово, Кемеровская область

АНАЛИЗ ЗАПИСЕЙ ПОЛЬЗОВАТЕЛЕЙ СОЦИАЛЬНЫХ СЕТЕЙ ДЛЯ ВЫЯВЛЕНИЯ АКТУАЛЬНЫХ ПРОБЛЕМ

***Аннотация:** в современном мире для политических движений важным аспектом построения программ является общественное мнение. Помимо прямых способов получить его можно на основе записей и комментариев в социальных сетях. Для автоматизации этого процесса нужна информационная система, которая будет автоматически собирать и обрабатывать данные для получения кластеров – тематик, актуальных в обществе. В статье выбирается метод проведения кластеризации, библиотека для программной реализации, а также показывается пример результата работы.*

***Ключевые слова:** анализ общественного мнения, анализ социальных сетей, компьютерная лингвистика, тематическое моделирование, нечёткая кластеризация, аддитивная регуляризация тематических моделей.*

Введение

В настоящее время для органов власти всех регионов очень важным является сбор и анализ информации о доминирующих настроениях в обществе для выбора направлений проведения внутренней и внешней политики. С целью проведения подобного анализа с декабря 2020 года по поручению президента [1] во всех субъектах РФ были созданы Центры управления регионами (ЦУР),

единые пункты мониторинга и обработки поступающих от жителей жалоб в разных сферах жизни региона. Кроме того, на основе поступающих в ЦУР данных формируют аналитические материалы, характеризующие социально-экономическую ситуацию в области, и вырабатывают предложения по дальнейшему развитию региона. Например, в Кемеровской области одним из способов формирования обращений является возможность отправить заявку с описанием проблемы на сайте «Кузбасс-онлайн» в специальном разделе, и эта заявка будет рассмотрена представителями администрации региона или соответствующего городского управления. Однако в самой структуре есть определённый недостаток: чтобы проблема стала известна правительству и была учтена в работе ЦУР, с ней должны обратиться к органам местного самоуправления. На практике очень небольшое количество граждан оставляет заявки с описанием проблем на тех ресурсах, которые ЦУР использует для мониторинга и анализа, или обращаются напрямую к представителям органов власти. По этой причине теоретически полезные разработанные механизмы оказываются неполноценными.

Потенциально полезным является полная модерация социальных сетей с целью обнаружения обсуждений актуальных вопросов за пределами официальных страниц представителей администрации, однако на практике для такого анализа необходимо вручную вычитывать комментарии, что включает в себя человеческий фактор, а также потребует слишком больших временных затрат. Альтернативный вариант – разработка информационной системы (далее – ИС), которая смогла бы в автоматическом режиме собирать данные и затем обрабатывать их с целью нахождения ключевых актуальных вопросов, поднимающихся среди населения. Такую систему можно разделить на несколько равнозначных составляющих, и одна из ключевых частей – механизм проведения анализа имеющихся данных, идущая за частью сбора данных. В принципе, в ней и решается сама задача – производится формирование тематик, для поиска которых система и может разрабатываться.

Нужно отметить, что сама идея анализа социальных сетей не является чем-то новым. Подобные системы часто используются в бизнесе с целью формирования маркетингового плана. Аналитики в компаниях таким образом изучают отзывы о своих товарах и услугах, встречаются нестандартные мнения, которыми с ними не делятся напрямую, и так далее. Часть таких систем даже доступны для использования в открытом доступе (примеры – Brand Analytics, Медиалогия, Babkee и пр.) Однако все они имеют свои недостатки, как правило, это либо высокая цена, либо не очень высокое качество кластеризации. Более того, поскольку все такие сервисы созданы в первую очередь для решения бизнес-задач, они могут собрать данные по определённому запросу (например, конкретный продукт или компания на рынке), однако для получения общей картины в них нет достаточно обширных аналитических функций. По этой причине возникает идея создания собственной системы, которая будет открытой и при этом сможет удовлетворять возникающие потребности и постоянно изменяющиеся конкретные цели (поиск внутри определённой тематики, например, экологии или политики). В общем случае алгоритм анализа текстовых данных (и, как следствие, работы приложения) следующий:

- считать полученные данные, отобрать из них те, которые будут использоваться в текстовом анализе;
- провести предобработку текстовых данных, включающую в себя:
 - лемматизацию – приведение слов к их начальным формам;
 - токенизацию – разбиение текста на слова и несловесные символы;
 - удаление несловесных символов – знаки препинания и эмодзи не нужны при анализе;
- приведение к нижнему регистру, необходимое для того, чтобы одно и то же слово в начале и в середине предложения не определялось как два разных;
- удаление стоп-слов, то есть слов, которые не несут смысловую нагрузку, но которые могут в больших количествах встречаться в текстах (например,

местоимения, служебные части речи и пр.). Можно использовать готовые шаблоны списков стоп-слов или создавать свои;

- провести тематический анализ;

- на основе анализа подсчитать количественные характеристики, например, определить вес тематики. Такие характеристики нужно определять после первых результатов в зависимости от тех параметров, которые приложение получало изначально;

- вернуть пользователю результаты работы. Поскольку в нашем случае при определённых параметрах результат может быть объёмным, необходимо сформировать общий отчёт, который сразу будет показан пользователю, и подробный, который изначально будет записан в базу. При необходимости пользователь может посмотреть детальную информацию по конкретной тематике или региону, однако изначально ему предпочтительнее получить общую картину.

Методы анализа текстовых данных

Основной смысл тематического анализа текстовых данных заключается в том, что по определённым приведённым к некоторому шаблонному формату входным данным нужно провести анализ, под которым может подразумеваться целый класс задач, и в итоге выдать некоторый ответ, которым, в зависимости от системы, может быть число или выражение, утверждение или алгоритм действий и так далее [2]. Причём как подготовка данных, так и анализ становятся тем сложнее, чем больше данных, и в случае анализа информации из социальных сетей данных будет очень много, хоть доля полезных из них может оказаться невысокой. Более того, поскольку необходимо всегда иметь актуальный список обсуждаемых в обществе тем, нельзя использовать заранее созданный шаблон-список тем, которые будут целью поиска. Таким образом, становится очевидным факт, что цель приложения – проведение кластеризации документов.

Кластеризация документов – это процесс, при котором все документы разбиваются на группы – кластера, объединённые какой-то одной тематикой.

Сам процесс кластеризации (как и практически любой другой процесс текстового анализа) делится на две важные части: разбиение текстовых данных на составляющие (слова, термы, вектора и пр.) и анализ результатов с помощью методов кластеризации (как правило, на основе расстояний). Методов и разбиения текстов, и кластеризации существует достаточно много, и они все детально разобраны в большом количестве публикаций [3]. В зависимости от поставленной задачи те или иные методы показывают лучшие результаты. В нашем случае одними из основных критериев являются большие объёмы данных, потенциально большое количество кластеров и их нечёткое определение, а также потенциально очень большое количество «мусорных» данных, которые так или иначе всё же нужно учитывать. Учитывая всё это, становится очевидно, что среди всех видов способов извлечения информации разумнее всего становится использование так называемого «Тематического моделирования», поскольку на выходе такие методы представляют наборы тематик, что в точности совпадает с тем, что требуется от приложения.

Тематическое моделирование (или тематический анализ) – метод статистического анализа, предназначенный для выявления тематик в наборе документов. В нашем случае тематический анализ поможет определить основные тематики, которые обсуждаются в обществе, и распределить их по популярности (частота встречаемости), то есть провести кластеризацию документов, причём необходимо использовать нечёткую кластеризацию (или вероятностное моделирование), так как некоторые сообщения могут относиться сразу к нескольким темам. Более того, за счёт имеющихся данных можно определять, является ли мнение общества по той или иной теме скорее положительным или отрицательным (для этого учитываются поставленные той или иной записи или комментарию лайки и репосты).

Почти всегда тематический анализ проводится при соблюдении некоторых стандартных условий:

- порядок документов, подаваемых на вход, не имеет значения;

- порядок слов в каждом из документов не имеет значения;
- слово в разных формах (например, склонения и падежи) считается одним и тем же словом;
- слова, которые часто встречаются во многих документах, не считаются полезными (часто они убираются из документов на этапе подготовки).

Тематическое моделирование – это метод, который детально разобран с математической точки зрения. В рамках кластеризации данных из социальных сетей все вышеуказанные условия выполняются, однако объём данных в теории может быть очень большим. К тому же необходимо учитывать, что текст в сообщениях в социальных сетях нередко может быть написан на разговорном языке с обилием жаргонизмов и других лексем, аналогично усложняющих работу по определению тематик. В итоге для получения приемлемых результатов необходимо использовать метод, который позволяет учитывать большое количество различных дополнительных параметров, связанных с текстовыми данными.

Помимо прочего стоит также отметить, что некорректно считать тему как какую-то реальную тему из нашей практики (вроде «экологии», «экономики», «спорта», «политики» и так далее). Такой вариант возможен только при заранее созданных матрицах тем и слов. Такие системы существуют и используются на практике, однако обычно они служат для генерации текстов. Для классификации имеющихся данных это невозможно (если необходимо получить качественный результат), поэтому мы получаем на выходе лишь словарь слов, объединённых как «Тема1», а обобщенное название нужно давать ей вручную.

Альтернативный вариант существует – это создание так называемых «корпусов», с помощью которых можно давать названия кластерам. Корпусом называется набор текстов-примеров, с которыми программа сравнивает описание получившихся кластеров на наличие соответствий в словах, термах и полных предложениях. Однако для создания качественных корпусов необходимо иметь полное представление о всех возможных словосочетаниях и

выражениях, чтобы покрыть как можно более качественно предметную область кластера, при этом не нагрузив лишними/дублирующими примерами. Для русского языка также характерна проблема многозначности слов, из-за чего сравнение слов из описания результата без контекста может сбить алгоритм. По этой причине создание удачных корпусов является нетривиальной задачей, однако совсем отбрасывать эту идею также не следует.

Методы тематического моделирования

Существует большое количество различных методов тематического моделирования. Самыми известными и популярными среди них можно назвать вероятностный латентно-семантический анализ и латентное размещение Дирихле.

Вероятностный латентно-семантический анализ предложен Томасом Хоффманом в 1999 г [4] как продолжение метода латентно-семантического анализа (LSA), изначально предложенного в 1990-м году [5]. Если говорить простым языком, основная идея LSA заключается в том, чтобы связать между собой разные по написанию (например, не однокоренные) слова из разных документов, которые, тем не менее, в данном конкретном случае будут близкими по смыслу. В качестве примера можно рассмотреть такие два выражения:

– *Атмосфера в городе N ухудшилась по сравнению с показателями предыдущего года.*

– *В N дышать уже просто невозможно, воздух отвратный!*

Лексика, используемая авторами, достаточно разная, поэтому выражения можно связать по названию города N, хоть мы и видим, что оба комментария не просто о городе N, но о плохом состоянии экологии. Однако компьютер не может это определить, так как слова не похожи. Метод LSA позволяет «обнаружить» эту латентное сходство и связать два комментария не только по городу N, но и по тематике экологии, придав схожие веса словам «атмосфера» и «воздух». Такие связи порождают пространство меньшей размерности, и таким образом формируются матрица документов-кластеров.

По своей структуре LSA часто сравнивают с простой нейронной сетью, в которой есть входной слой – слой слов (термов), выходной слой – слой документов, и слой весов, связывающий входной и выходной слои.

Вероятностный латентно-семантический анализ (pLSA) отличается от обычного тем, что LSA имеет очевидный недостаток: слова в любом языке часто являются многозначными, поэтому не всегда можно определить тему без детального понимания контекста (который в случае сообщений в социальных сетях может прослеживаться вообще в другом сообщении). Вероятностная модель появления пары «документ»–«слово» может быть записана несколькими способами, например, таким:

$$p(d, w) = \sum_{t \in T} p(d)p(w|t)p(t|d)$$

Здесь w – некоторое слово из массива W всех слов во всех документах, t – некоторая тема из списка тем T , d – некоторый документ из списка документов D . Следовательно, каждый документ d описывается неизвестным распределением $p(t|d)$, $t \in T$, каждая тема t – распределением $p(W|t)$ $w \in W$. pLSA относит все «важные» слова в сообщениях к разным темам t с некоторой вероятностью, что упрощает дальнейшую интерпретацию результатов работы метода.

Однако и pLSA не лишён недостатков. Основными из них можно назвать вероятность переобучения модели, так как количество слов растёт с каждым новым добавленным документом, и следующей из этого невозможностью по той же модели строить прогноз при добавлении нового документа, из-за чего необходимо перестраивать всю модель заново, что особенно актуально в случае отсутствия стандартного формата написания комментариев в социальных сетях.

Ещё одним популярным методом тематического моделирования является метод NMF, предложенный в 2003 году [6]. Его отличие от LSA в том, что матрицы, на которые раскладываются изначальная матрица данных, здесь не всегда полученное пространство будет ортогональным. Таким образом, качество

кластеризации должно быть выше. Однако на практике это скорее позволяет лишь сделать кластера более различными, не влияя на качество кластеризации.

Метод латентного размещения Дирихле (LDA), предложенный Дэвидом Блеем в 2003 г., построен с учётом недостатков метода pLSA [7]. Сам метод построен на той же вероятностной модели, что и pLSA, однако в LDA считается, что вектора документов $p(t|d)$ порождаются одним и тем же вероятностным распределением на нормированных векторах размерности $|T|$; аналогично для векторов тем $p(w|t)$ и векторов размерности $|W|$. Основная идея заключается в том, что каждый документ представляет собой набор нескольких тем, и каждое слово в документе можно отнести к некоторой теме. При этом, если запись строго относится к одной конкретной теме, это указать очень просто: выходной вектор-тем будет состоять из нулей и одной единицы, соответствующей нужной теме. Кроме того, метод позволяет строить оценку для документов, не входивших в обучающую выборку, используя алгоритм вариационного вывода.

Метод LDA является хорошим стартом для развития дальнейших методов, хоть и сам по себе достаточно эффективен. Однако в жизни задача тематического моделирования имеет большое количество разных решений, и LDA предоставляет лишь одно из них. Поэтому под разные задачи строятся различные расширения, основанные на методе LDA и нуждах конкретного задания.

Одним из таких расширений является метод аддитивной регуляризации тематических моделей (ARTM). Вообще для получения более конкретных результатов для метода LDA строятся так называемые регуляризаторы, задающие свойства модели в виде критериев. ARTM позволяет сразу применять к модели несколько регуляризаторов. Например, при анализе комментариев к записи, можно учитывать тематику самой записи и группы, в которой она находится. Следовательно, можно добавлять новые темы и подтемы по необходимости. Также среди регуляризаторов ARTM есть, в частности, механизмы, позволяющие учитывать не только текст сообщения, но и, например, метаданные – автора, время публикации и так далее.

Таким образом, оценка принадлежности комментария методом LDA к некоторой теме будет вероятностной, поскольку комментарий может затрагивать сразу несколько тем, что не позволял делать pLSA. Кроме того, при выборе ARTM можно учитывать метаданные комментария, чтобы не выстраивать вручную связи с тематикой группы и записи, а также анализировать другие сообщения того же пользователя для выстраивания более полной картины.

Помимо этого, на основе метода ARTM есть библиотека BigARTM [8] (big – большой), которая поддерживает механизмы распараллеливания входных данных и позволяет вести работу с большими объёмами данных без потерь в качестве оценок. За счёт этого можно сократить время анализа. Таким образом, использование библиотеки BigARTM является разумным выбором среди алгоритмов кластеризации. Дополнительно стоит отметить, что библиотека поддерживает кластеризацию не только методом ARTM, но и LDA, следовательно, можно проводить кластеризацию обоими методами и сравнивать полученные двумя способами кластера.

Разработка и тестирование

Для разработки приложения для кластеризации разумнее всего использовать язык Python, поскольку на нём реализовано большое количество готовых решений по обработке текста. К тому же, библиотека BigARTM имеет поддержку языка Python и документацию по работе через этот язык программирования.

Однако, как и при работе с любой другой библиотекой машинного обучения, здесь нужно провести подготовку исходных данных перед тем, как запускать процесс [9]. В общем случае алгоритм работы с BigARTM является таким.

1. Загрузить все необходимые библиотеки, а также данные, полученные из социальной сети «ВКонтакте» (или любого другого источника, но в примере использовалась именно эта социальная сеть), сформировать список стоп-слов.

2. Провести подготовку записей и комментариев: удалить знаки препинания, стоп-слова, смайлики и другие ненужные символы (в частности,

можно удалить обращения вида “<имя автора комментария, на который данный комментарий является ответом>, ” в ответах на комментарии), после чего проверить пустые строки (при обнаружении удалить), а также провести лемматизацию – привести все слова к начальной форме.

3. Привести исходные данные в необходимый для алгоритма вид «vowpal wabbit», то есть мешок слов. В нём в каждой строке находятся слова, употребляемые в документе (то есть, в записи или комментарии).

4. Разбить данные на небольшие пакеты (так называемые batch-и), а также сформировать словарь, включающий в себя все слова из всех текстов (он будет необходим для кластеризации).

5. Провести кластеризацию методом BigARTM. Здесь после окончания кластеризации нужно провести анализ результатов. Поскольку в методе BigARTM нужно вручную указывать некоторые важные параметры (в частности, количество итераций и выходных тем), проводить кластеризацию придётся много раз. Сравнивая результаты, можно подобрать наиболее оптимальные параметры.

6. Поскольку нередко тот или иной комментарий нельзя отнести строго к какой-то одной теме, в BigARTM можно посмотреть вероятностную оценку отношения записи/комментария к той или иной теме.

На первом этапе производился сбор данных. Вообще для этих целей в полноценной ИС следует использовать отдельный серверный блок – приложение, зарегистрированное по правилам «ВКонтакте» так, чтобы можно было полноценно использовать существующий api. Однако в конечном счёте нет большой разницы, как и в каком виде были получены данные. Это влияет только на первый этап, когда необходимо получить тексты записей и комментариев. В дальнейшем будем считать, что данные были получены по api «ВКонтакте» в формате json.

Подготовка получившихся текстовых данных может быть разделена на несколько этапов, описанных в пункте 2 алгоритма выше. Отдельно стоит

отметить только список стоп-слов. Здесь наиболее оптимальным будет взять существующий базовый список стоп-слов (например, из всё той же библиотеки `nlTK`, которая уже использовалась в других функциях предобработки) и затем дополнять его в рамках тестовых кластеризаций, просматривая результаты работы.

Выбранная библиотека `BigARTM` позволяет представить данные в разном формате. В зависимости от указания параметров можно, к примеру, просмотреть вероятностные оценки отношения каждого документа к каждому кластеру. Таким образом, например, можно подсчитать количество сообщений в каждом кластере (выбирая тот кластер, у которого наибольший вес). Чтобы определить сами кластера, можно использовать метод отображения n ключевых слов (где n – число, задаваемое пользователем) [10]. Результатом работы модуля в таком формате будет список слов, объединённых в кластера, с указанием (при необходимости) веса для каждого слова (аналогично документам, можно просмотреть оценки отношения каждого слова в мешке слов к каждому кластеру, однако эти данные в полном виде будут очень большими, а потому имеет смысл лишь поиск конкретного слова и его набор вероятностей).

Рассмотрим пример разбиения тестовых данных на три кластера. Результат кластеризации представим в виде списка ключевых слов. Получились следующие результаты:

- `topic_0`: [водитель, автобус, ДТП, автомобиль, район, движение, кемеровчанин, мужчина, летний, получать];
- `topic_1`: [график, компания, рубль, требоваться, условие, обучение, требование, магазин, плата, продажа];
- `topic_2`: [цена, рубль, сергей, житель, узнавать, долг, цирк, отмечать, получать, билет].

Можно заметить, что получившиеся кластера хорошо обобщаются. Первая тематика – дорожное движение, вторая – вопросы, связанные с трудоустройством, а третья – досуг и сфера услуг. Стоит отметить слово «сергей» в третьем кластере.

С одной стороны, они совершенно тут не нужны и его можно было бы добавить в список стоп-слов. Однако Сергей – это имя губернатора Кемеровской области, следовательно, тот факт, что именно это имя оказалось в списке ключевых слов где-либо, может быть также интересным. Проверить правильность таких кластеров на практике очень трудно, поскольку для кластеризации полноценных метрик качества нет. Всё сильно зависит от данных, и подогнать под них какие-то критерии сложно. Потенциальным вариантом получения «правильных» ответов может быть разве что проведение других видов социологических анализов и сравнение получившихся результатов [11].

Ещё одним важным результатом, который можно получить после проведения кластеризации, является θ -матрица. С её помощью можно посмотреть вероятностную оценку соответствия каждого документа каждому кластеру. Используя её, можно подсчитать количество документов в каждом кластере, таким образом определив вес каждой тематики (каждую запись относить к тематике, вероятностная оценка соответствия которой наиболее высока). Как следствие, так можно выяснить, какая из тематик наиболее актуальна среди сформированных. В нашем примере соотношение записей было таким:

- topic0: 2447;
- topic1: 1405;
- topic2: 1674.

В итоге первая тематика оказалась наиболее актуальной. Это справедливо, поскольку в г. Кемерово, по которому производился сбор тестовых данных, за охватываемый период (сбор производился по 500 записей с нескольких самых больших по числу подписчиков страниц) было произведено большое количество дорожных работ, что часто приводило к перекрытым дорогам.

Выводы

Для получения представления о настроениях в обществе можно использовать данные, имеющиеся в социальных сетях, и с помощью алгоритмов машинного обучения проводить анализ текущей ситуации. В данной статье были разобраны

самые популярные методы – вероятностный латентно-семантический анализ, NMF и латентное размещение Дирихле – и разобраны их недостатки для данной задачи, а также аддитивной регуляризации тематических моделей, использование которого должно позволить получить информацию о картине настроений в обществе без больших затрат времени и человеческого труда. Также был разобран пример работы с библиотекой и пример результата кластеризации.

Список литературы

1. Перечень поручений по итогам заседания Совета по развитию местного самоуправления от 1 марта 2020 года. Пр-354, п. 3 [Электронный ресурс]. – Режим доступа: <http://www.kremlin.ru/acts/assignments/orders/62919>
2. Шокин Ю.И. Проблемы поиска информации / Ю.И. Шокин, А.М. Федотов, В.Б. Баракнин. – Наука, 2010. – ISBN 918–5-02–018969–0.
3. Пархоменко П.А. Обзор и экспериментальное сравнение методов кластеризации текстов / П.А. Пархоменко, А.А. Григорьев, Н.А. Астраханцев // Труды ИСП РАН. – 2017. – Т. 29. Вып. 2. – С. 161–200. DOI: 10.15514/ISPRAS-2017–29(2)-6. – EDN YNEYWH
4. Hofmann Thomas Probabilistic latent semantic indexing. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. ACM. 1999, pp. 50–57.
5. Scott Deerwester, Susan T Dumais, George W Furnas et al. Indexing by latent semantic analysis. Journal of the American society for information science. – 1990. – Т. 41. №6. – pp. 391.
6. Xu Wei, Liu Xin, Gong Yihong. Document clustering based on non-negative matrix factorization. Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM. 2003, pp. 267–273.
7. Blei David M, Ng Andrew Y, Jordan Michael I. Latent dirichlet allocation. Journal of machine Learning research. – 2003. – Т. 3. – № Jan. – pp. 993–1022.

8. BigARTM – тематическое моделирование на Python [Электронный ресурс]. – Режим доступа: <https://newtechaudit.ru/bigartm-tematicheskoe-modelirovanie-na-python/>
9. Воронцов К.В. Вероятностное тематическое моделирование: теория регуляризации ARTM и библиотека с открытым кодом BigARTM / К.В. Воронцов. – Московский государственный университет им. М.В. Ломоносова, Московский физико-технический институт (государственный университет). Федеральный исследовательский центр «Информатика и управление» РАН, 2023 г. – ISBN 978–5-9710–9933–8.
10. Руководство пользователя библиотеки BigARTM [Электронный ресурс]. – Режим доступа: <https://bigartm.readthedocs.io/en/stable/tutorials/index.html>
11. Кранзеева Е.А. Региональное социально-политическое взаимодействие: субъекты и репрезентанты (на основе Кузбасса) / Е.А. Кранзеева, А.Л. Бурмакина, О.А. Алтемерова // Вестник РГГУ. Серия: Философия. Социология. Искусствоведение. – 2020. – №3. – ISSN 2073–6401. – DOI 10.28995/2073-6401-2020-3-76-86. – EDN VVKKBM