Тематическое моделирование научных статей

Бондаренко Даниил

Март 2023

Аннотация

Курсовая работа посвящена исследованию применения методов машинного обучения для анализа текстовых данных научных статей. В работе рассматриваются основные понятия и методы тематического моделирования, а также их применение для анализа научных статей. Основной целью курсовой работы является выявление тематических групп в научных статьях по 6 областям исследований. Для достижения этой цели для некоторых моделей была проведена предварительная обработка данных, включающая стемминг текстов и удаление стоп-слов. Затем была произведена кластеризация статей на тематические группы на основе классических подходов к тематическому моделированию (LDA, NFM), и более современных архитектур, таки как BERTopic, Top2Vec. В рамках курсовой работы также были проанализированы результаты кластеризации, оценены тематические группы на соответствие действительности и проведен сравнительный анализ методов тематического моделирования. Результаты курсовой работы могут быть использованы для более эффективного поиска и анализа научных статей в выбранной области исследования, а также для выявления тенденций и трендов в научной литературе.

Проект доступен по ссылке: https://github.com/daniilbond1/Topic-modeling-for-article-titles-

1. Вступление

Понимание содержания и тематики научных статей является важной задачей в различных областях, таких как классификация текстов, поиск информации, рекомендательные системы и многие другие. Тематическое моделирование – это мощный инструмент машинного обучения, который позволяет автоматически выделять темы из больших корпусов текстов. Это позволяет найти наиболее вероятные темы для текстовых документов. Таким образом, тематическое моделирование может быть использовано для автоматического присваивания тематических меток научным статьям, что позволит более быстро и точно определить тему исследования.

В данной работе было проведено сравнение классических и нейросетевых подходов к тематическому моделированию. Нейросетевые методы обычно используют глубокое обучение для выделения признаков и обучения модели, что может давать более точные результаты, но также требует больше вычислительных ресурсов. С другой стороны, более классические подходы, такие как Latent Dirichlet Allocation (LDA) и Non-negative Matrix Factorization (NMF), являются более простыми в реализации и, тем не менее, могут быть эффективными при работе с большими объемами текстовых данных.

2. Обзор литературы

NLP (Natural Language Processing) – одно из направлений искусственного интеллекта, которое занимается анализом и синтезом естественного языка. Тематическое моделирование (topic modelling) является подходом, использующимся в NLP для

автоматической кластеризации текстовых данных путем выявления тем, в документах или текстовых фрагментах. Методы NLP и тематического моделирования используются для решения различных задач, таких как автоматическая разметка текстов по их тематикам, кластеризация новостных статей, определение настроения в текстах отзывов и комментариев, а также для анализа социальных сетей. В данной работе будут рассмотрены различные методы тематического моделирования и реализованы некоторые из них (LDA, NFM, BERTopic, Top2Vec) для выявления тематик научных статей по их названиям.

Метод мешка слов (Bag of Words) – один из самых простых методов тематического моделирования. В этом методе каждый документ представляется в виде набора слов, которые появляются в нем без учета грамматических связей между ними. Затем на основе статистического анализа частоты встречаемости слов в документах, строится матрица, которая позволяет определить темы, наиболее часто встречающиеся в коллекции документов. Таким образом, метод мешка слов помогает выделить наиболее значимые темы в текстовых данных. Однако этот метод может быть не точным из-за того, что он не учитывает контекст и порядок слов.

Более точный метод тематического моделирования – это Latent Dirichlet Allocation (LDA). В основе LDA лежит модель, которая представляет каждый документ как распределение набора скрытых тем, а каждую тему - как распределение набора скрытых слов. На основе частотности слов в документах LDA может определить темы и слова, связанные с этими темами. LDA позволяет получать различные параметры, такие как вероятность слова в конкретной теме, вероятность темы в конкретном документе и вероятность слова в конкретном документе.

Метод LDA (Latent Dirichlet Allocation) был предложен в 2003 году Д. Блеем, Э. Нгом и М. Джорданом из Университета Беркли [1]. В своей работе "Latent Dirichlet Allocation" они описали метод, который позволяет моделировать темы в текстах, используя вероятностную модель. Метод LDA получил широкое распространение в области анализа текстов и машинного обучения и используется для различных задач, таких как классификация документов, поиск информации и анализ тональности. LDA позволяет анализировать не только одиночные новости, но и большие объемы текстовых данных, такие как новостные потоки или коллекции предметно-ориентированных статей.

Алгоритм LDA имеет следующие шаги:

1. Инициализация - задание случайных значений для параметров модели
2. Присвоение словам в каждом документе некоторой случайной начальной темы
3. Повторение следующих двух шагов до тех пор, пока не будет достигнута определенная точность:
    - распределение тем в документах пересчитывается и обновляется на основе слов, которые были присвоены этим темам;
    - распределение тем в корпусе (всех документов) пересчитывается и обновляется на основе новых распределений тем в документах.

В результате работы алгоритма LDA получаются матрица распределения тем по документам и матрица распределения слов по темам. Эти матрицы можно использовать для определения наиболее важных тем и слов в наборе данных.

На основе LDA были созданы такие методы, как Hierarchical Dirichlet Process (HDP) [2], Dynamic Topic Models (DTM) [3] и Correlated Topic Models (CTM) [4]. HDP является расширением LDA, в котором количество тем не требуется определять заранее. HDP

использует более гибкую байесовскую модель, которая позволяет находить бесконечное количество тем в каждом документе. Это означает, что каждый документ может иметь уникальные темы, которых еще не было в других документах. Таким образом, HDP является более гибкой версией LDA, которая позволяет моделировать бесконечное количество тем и учитывать возможность наличия уникальных тем в каждом документе.

DTM также является расширением метода LDA. Его отличие состоит в том, что он позволяет анализировать динамические изменения в структуре тем во времени. Он использует гибкую байесовскую модель и моделирует каждый временной интервал как отдельный набор документов. На каждом интервале обучается LDA-модель, которая позволяет выявить изменения в структурах тем, появление новых и исчезновение старых тем. Это является преимуществом по сравнению с самим методом LDA, который используется в статических данных.

CTM также основан на LDA. Его отличие состоит в возможности учёта корреляции тем или слов в каждой теме. В CTM каждая тема описывается распределением вероятностей на словах и распределением вероятностей на корреляции между темами. Для поиска корреляций между темами в CTM используется модель гауссовской связи, которая определяет связь между каждой парой тем в модели. Каждая тема в модели представляется как многомерное гауссовское распределение, и более высокая корреляция между двумя темами соответствует большему перекрытию между их гауссовскими распределениями. Таким образом, CTM позволяет моделировать зависимости между темами с помощью различных распределений вероятностей. Хотя LDA и CTM имеют сходные основные концепции, CTM более продвинут и может быть полезнее в некоторых приложениях, где важны связи и зависимости между темами, например, при анализе социальных сетей или общественного мнения.

Другим подходом тематического моделирования является использование матричных разложений. Например, Latent Semantic Analysis (LSA) - метод, который использует сингулярное разложение матрицы слов (SVD) для извлечения тематических компонент [5]. LSA позволяет снизить размерность исходной матрицы, устранить шум и найти скрытые закономерности в данных. Существует также модификация LSA – Probabilistic Latent Semantic Analysis (pLSA) [6]. Он относится к вероятностным подходам, так как использует байесовские методы для определения вероятности слов в темах. pLSA позволяет учитывать контекст, в котором используются слова, путем группировки слов схожих тематик, что позволяет получить более точное моделирование тем.

Помимо LSA, к группе матричных методов относится NMF (Non-negative Matrix Factorization) [7]. Это метод линейной алгебры для разложения матрицы на две неотрицательные матрицы, которые вместе приближают исходную матрицу наилучшим способом. Алгоритм итеративно изменяет значения матриц A и B таким образом, чтобы их произведение приближалось к исходной матрице X. При этом сохраняется структура исходных данных, а также гарантируется неотрицательность как базисных элементов, так и весов.

Однако для успешного применения NMF необходимо правильно инициализировать начальное значение, иначе это может привести к получению неверных выводов. NMF может использоваться в качестве шага предварительной обработки для уменьшения размерности в задачах классификации, кластеризации, регрессии и других.

Кроме того, этот метод может быть использован в любой ситуации, когда матрица входных данных не имеет отрицательных элементов.

Для тематического моделирования также существуют подходы, основанные на глубоком обучении. Одним из них является TopicRNN (Topic-Aware Neural Language Modeling). В нем используется двухслойная рекуррентная нейронная сеть с механизмом внимания (attention mechanism) и векторное представление слов для определения наиболее вероятных тем для каждого документа [8]. Далее, модель определяет вероятности принадлежности каждого слова к различным темам. Для этого используется softmax-слой на выходе модели. Чтобы обучить модель, используется метод обратного распространения ошибки (Backpropagation). В другом подходе для этих же целей используются иерархические нейронные сети (Hierarchical Attention Network) с механизмом внимания [9].

Помимо вышеописанных, существуют гибридные методы тематического моделирования, которые комбинируют различные инструменты и подходы для получения более точных результатов. Например, комбинация метода генерации эмбеддингов Doc2Vec, позволяющего получать векторные представления для целых документов, и алгоритмов кластеризации. Метод Doc2Vec был предложен в 2014 году в статье "Distributed Representations of Sentences and Documents" авторов Quoc Le и Tomas Mikolov [10]. В этой статье также рассматривается использование Doc2Vec для кластеризации документов.

Наиболее популярным среди гибридных методов можно назвать Top2Vec. Top2Vec использует предобученную модель Doc2Vec для генерации эмбеддингов документов, которые затем используются для кластеризации с помощью алгоритма кластеризации DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [11]. Для каждого кластера рассчитываются тематические вектора и сравниваются их с векторами слов, которые присутствуют в каждом предложении в данном кластере. Таким образом, каждый кластер имеет связанные с ним темы, состоящие из наиболее важных слов. Несколько наиболее важных тем определяются для каждого документа на основе взвешенного среднего их тематических векторов, где веса зависят от степени сходства векторов предложений. Top2Vec автоматически определяет количество тем внутри данных и организует документы в кластеры с семантически схожим содержанием, что делает его быстрым и высокоэффективным методом тематического моделирования.

Дополнительно стоит упомянуть BERTopic, использованный в данной работе [12]. Это метод тематического моделирования использует предобученную модель BERT в качестве основы для извлечения семантических тем в тексте.

При анализе текста BERTopic решает следующие задачи:

1. Предварительная обработка текста: тексты очищаются от стоп-слов и стемминга, чтобы упростить его представление.
2. Извлечение векторных представлений на основе предобученной модели BERT и алгоритма encoding.
3. Создание графа, используя эти эмбеддинги. Граф строится на основе косинусного подобия между векторами, где каждый вектор представляет документ, а ребра между векторами представляют их семантическую связь. Документы, которые сильно связаны друг с другом, объединяются в топики (темы) в графе.

4. Выделение топиков (тем) из графа путем кластеризации его с помощью алгоритма Leiden. Leiden – это алгоритм, который используется для разделения графа на несколько тематических кластеров, чтобы выделить основные темы в тексте.

5. Наконец, производится тонкая настройка тематической модели, путем усложнения критериев косинусного подобия, чтобы повысить точность определения тем.

BERTopic построен на основе BERT и позволяет получать высокоуровневые темы, понятные для конечного пользователя. Он показал высокую точность на различных наборах данных, включая новостные статьи и социальные сети. Он также позволяет легко определять доминирующие темы в данных, что делает его полезным для анализа тематики больших объемов текстовых данных.

В заключение стоит отметить, что современные методы тематического моделирования имеют большой потенциал в обработке текстовых данных разного типа. Эти методы позволяют выявлять скрытые темы и находить зависимости, которые не видны при поверхностном анализе данных. Их использование может быть очень полезным в различных областях, таких как маркетинг, социальные исследования, научные публикации и т. д.

3. Описание использованных моделей

3.1. Модель латентного размещения Дирихле (LDA)

LDA — модель тематического моделирования, которая использует байесовскую вероятность для разделения документа на темы. На первом шаге устанавливается определенное количество тем k. Каждая тема определяется мультиноминальным распределением всех слов в корпусе. Распределение тем соответствует распределению Дирихле – небинарному распределению вероятностей. LDA генерирует темы согласно следующему алгоритму:

1. Устанавливается k количество тем
2. Генерируется случайным образом распределение тем документа $\beta \sim$ Дирихле для каждого документа
3. Случайным образом генерирует распределение тем-слов $\theta \sim$ Дирихле
4. Для каждого слова w в каждый документ d:
    - случайным образом выбирает тему k, учитывая распределение тем документа
    - присваивает слово w теме j, учитывая

$$p(w, j) = p(k|d) * p(w|k)$$

После этого документы представляются как темы смеси и темы представлены в виде смесей слов. Генерация случайного распределения тем документа означает, что каждое слово в документе сначала назначается теме случайным образом. Затем для каждого слова в документе случайным образом выбирается тема с учетом распределения тем документа. Он вычисляет вероятность выбранной темы с учетом распределения тем документа. После этого вычисляется вероятность появления наблюдаемого слова в выбранной теме. Слово, наконец, назначается теме, в которой слово, скорее всего, появится, учитывая произведение двух предыдущих вероятностей. С каждым документом слова становятся более связанными с определенными темами, чем с другими.

3.2. Метод неотрицательной матричной факторизации (NMF)

Неотрицательная матричная факторизация (NMF) является методом линейной алгебры, который применяется для разложения неотрицательной матрицы на произведение двух матриц более низкого ранга. Этот алгоритм обучения без учителя используется в анализе текстов и изображений, а также в рекомендательных системах. Его основное применение заключается в том, чтобы уменьшить размерность данных в пространстве меньшей размерности, что позволяет сохранить ценную информацию о структуре данных.

Алгоритм NMF состоит из следующих шагов:

1. Инициализация неотрицательных матриц: Матрицы W и H инициализируются случайно, и каждый элемент должен быть неотрицательным для получения неотрицательных матриц.
2. Вычисление приближения: Произведение матриц W и H дает приближение исходной матрицы. Алгоритм сравнивает исходную матрицу и приближение и вычисляет ошибку, которая оценивает, насколько хорошо матрицы W и H приближают исходную матрицу.
3. Обновление матриц: Матрицы W и H обновляются с помощью градиентного спуска или других оптимизационных методов, чтобы уменьшить ошибку приближения. Обновление происходит по очереди: сначала обновляется матрица H, затем матрица W и так далее.
4. Повторение шагов 2 и 3 до сходимости.

В контексте тематического моделирования каждый элемент матрицы A соответствует количеству появлений слова в документе, а каждый элемент матриц W и H представляет вероятность вхождения слова в тему и вероятность вхождения темы в документ соответственно. Результатом NMF являются темы (кластеры слов) и распределение этих тем на документы, что позволяет понимать, о чем говорит каждый документ в коллекции и какие темы и в какой мере входят в документы.

3.3. Модель Top2Vec

Алгоритм top2vec состоит из трех основных этапов:

1. Предварительная обработка: текстовые документы преобразуются в векторы признаков, используя алгоритмы векторизации (например, TF-IDF).
2. Кластеризация: кластеры документов создаются на основе сходства векторов признаков. Top2vec использует алгоритм HDBSCAN для создания кластеров, который позволяет одновременно определить количество кластеров и определить выбросы (документы, которые не относятся к какому-либо кластеру).
3. Тематическое моделирование: на основе кластеров, top2vec строит уникальные темы для каждого кластера. Он использует алгоритм LDA (Latent Dirichlet Allocation) для построения тем. Каждая тема представлена набором ключевых слов, которые наиболее характерны для данного кластера.

В результате работы top2vec пользователь получает список тем (группы документов), а также список ключевых слов для каждой темы. Это позволяет использовать top2vec для кластеризации и категоризации больших коллекций документов, а также для анализа тенденций и популярных тем в текстовых данных.

3.4. **Модель BERTopic**

BERTopic — это модель тематического моделирования, которая использует BERT для создания тем. Он преобразует документы в эмбеддинги, уменьшает размерность и группирует их для извлечения тем с использованием модифицированной версии TF-IDF. Он использует SBERT, который использует предложения для получения эмбеддингов документов, чтобы учесть семантику. UMAP используется для уменьшения размерности, так как сохраняет локальные и глобальные признаки лучше, чем другие методы, и повышает производительность алгоритмов кластеризации.

UMAP снижает размерность данных за счет построения высокомерного представления данных при помощи "fuzzy simplicial complex" графа, где каждая точка данных соединяется с другими точками, где радиусы пересекаются. Затем UMAP выбирает локальное значение радиуса для каждой точки, что позволяет сбалансировать количество соединений. После построения высокомерного графа UMAP приближает его к графу меньшей размерности.

Эмбеддинги с пониженной размерностью кластеризуются с помощью HDBSCAN, который использует плотность для кластеризации точек данных на основе расстояния взаимной досягаемости. Алгоритм измеряет плотность с помощью "основного" расстояния, которое представляет собой расстояние между данной точкой и ее самым дальним соседом в пределах k ближайших соседей. Кластеры формируются на основе пороговой плотности, то есть группа точек считается кластером только в том случае, если их плотность выше определенного порога. Заключительным шагом является иерархическое добавление к DBSCAN.

HDBSCAN избегает плохих кластеров, вызванных различной плотностью, рассматривая плотность как «гору» с несколькими пиками. Глобальные пороговые значения могут привести к плохим кластерам, если количество точек в базе больше пиков или наоборот. HDBSCAN решает эту проблему путем разделения кластеров на основе локальной плотности. Это автоматизирует порог кластеризации и приводит к улучшению кластеров, которые служат основой для тем.

Перед тем, как кластеры можно будет использовать в качестве тем, необходимо сделать последний шаг: создать распределение между темами и словами. BERTopic делает это с помощью функции TF-IDF. Tf-IDF сама по себе является модифицированной матрицей количества терминов и определяется как:

$$W_{t,d} = t\,f_{t,d} \cdot log(\frac{N}{df_t})$$

BERTopic применяет TF-IDF к кластерам документов для создания разделов. В этом случае кластеры можно рассматривать как корпус, а каждый кластер как единый документ. Для этого все документы в кластере объединяются вместе. Результатом является кластерный TF-IDF, который измеряет важность слов в кластерах, а не в отдельных документах. Конечный продукт представляет собой распределение темы и слов для каждого кластера документов.

4. Описание экспиремента

4.1 Описание датасета

В данной работе использовался набор данных, взятый из соревнования на платформе kaggle. Он состоял из названий научных статей, разделенных на 6 тем:

- Computer Science
- Physics
- Mathematics
- Statistics
- Quantitative Biology
- Quantitative Finance

Так как целью данной работы является разделение названий научных статей по тематикам, метки тем из дататасета были удалены и в дальнейшем обучении не использовались. Также стоит отметить, что некоторые научные статьи могут принадлежать сразу к нескольким темам. Мы будем допускать, что таких ситуаций достаточно мало, поэтому не будем делать на это поправку при оценке результатов работы.

4.2 Метрик

В данной работе для оценки качества будем использовать метрику coherence - это метрика качества тематической модели, которая измеряет степень семантической связности терминов внутри каждой темы. Более конкретно, coherence оценивает, насколько хорошо термины, относящиеся к одной теме, "связаны" друг с другом по смыслу.

Coherence базируется на анализе значимости "словосочетаний" или "n-грамм" (наборы из двух или более слов) в каждой теме. Она использует подход, основанный на сравнении соответствия распределения словосочетаний внутри темы и некой референтной коллекции текстов.

Значения coherence могут находиться в интервале от -1 до 1. Положительные значения означают более логичные и связные темы, отрицательные - наоборот. Чем ближе значение к единице, тем лучше качество модели.

Coherence является одной из наиболее распространенных метрик для оценки тематических моделей и используется в различных приложениях, включая анализ текстов, поиск информации и аналитику социальных медиа.Coherence - это метрика, которая измеряет когерентность тематической модели на основе взаимосвязи между терминами, содержащимися в темах. Для расчета coherence используется статистический анализ n-грамм, которые часто встречаются в текстовых данных.

Одним из наиболее распространенных методов расчета coherence является подход, основанный на PMI (pointwise mutual information), который вычисляет степень связности между двумя терминами в контексте текста. PMI выражается следующей формулой:

PMI(w_i, w_j) = log ((P(w_i, w_j) / P(w_i) * P(w_j)))

где w_i и w_j - термины, P(w_i, w_j) - вероятность встретить термины w_i и w_j вместе в тексте, P(w_i) и P(w_j) - вероятность встретить каждый из терминов w_i и w_j в тексте.

В методе расчета coherence на основе PMI, для каждой темы вычисляется среднее значение PMI между всеми парами терминов, содержащихся в теме:

PMI(T) = (1/|T| * (|T|-1)) * ∑∑ PMI(w_i, w_j)

где T - множество терминов в теме, |T| - количество терминов в теме, w_i и w_j - термины в множестве T.

Для более удобного интерпретирования значения coherence, обычно используется Normalized Pointwise Mutual Information (npmi), который вычисляется как:

npmi(T) = (2 / (|T| * (|T|-1))) * ∑∑ log ((P(w_i, w_j) / P(w_i) * P(w_j))) / -log(P(w_i, w_j))

npmi дает возможность сравнивать coherence между различными темами или моделями, поскольку он шкалируется в диапазоне от -1 (полное отсутствие связи между терминами) до 1 (идеальная связность терминов в теме). Высокое значение npmi свидетельствует о том, что термины в теме сильно связаны семантически, что говорит о хорошей когерентности тематической модели.

5. Результаты работы

5.1. LDA

В данном примере алгоритм LDA применялся для анализа тематики статей из разных областей знаний. Результаты работы алгоритма показывают, что темы названий статей нередко пересекаются и в некоторых названиях содержится информацию о нескольких темах.

```
[(0,
  '0.013*"imag" + 0.012*"data" + 0.011*"use" + 0.011*"learn" + 0.009*"social" + 0.009*"detect" + 0.008*"predict" +
0.007*"toward" + 0.007*"analysi" + 0.006*"dark"'),
 (1,
  '0.050*"network" + 0.039*"learn" + 0.025*"model" + 0.020*"neural" + 0.019*"deep" + 0.017*"use" + 0.013*"gener" +
0.010*"graph" + 0.009*"data" + 0.008*"analysi"'),
 (2,
  '0.011*"studi" + 0.011*"system" + 0.008*"control" + 0.007*"robot" + 0.006*"model" + 0.005*"stabil" + 0.005*"synthesi" +
0.005*"machin" + 0.005*"separ" + 0.004*"translat"'),
 (3,
  '0.017*"estim" + 0.013*"optim" + 0.011*"space" + 0.010*"stochast" + 0.010*"model" + 0.009*"distribut" +
0.008*"function" + 0.008*"problem" + 0.008*"applic" + 0.007*"algorithm"'),
 (4,
  '0.011*"optim" + 0.010*"system" + 0.008*"comput" + 0.008*"use" + 0.008*"matrix" + 0.008*"model" + 0.008*"power" +
0.008*"process" + 0.007*"problem" + 0.007*"effici"'),
 (5,
  '0.013*"quantum" + 0.011*"field" + 0.010*"equat" + 0.009*"magnet" + 0.009*"group" + 0.008*"dynam" + 0.008*"algebra" +
0.007*"theori" + 0.007*"phase" + 0.007*"model"')]
```

Исходя из результатов алгоритма, можно предположить, что:

1. Тема 1 связана с Математикой, а также частично с Вычислительной биологией.
2. Тема 2 имеет большую связь с Физикой и Финансовой математикой.
3. Документ 3 также связан с Математикой, но также имеет элементы Статистики и Финансовой математики.
4. Документ 4 сосредоточен на Компьютерных науках и Статистики, но также содержит элементы Математики и Вычислительной биологии.
5. Документ 5 связан в основном с Вычислительной биологией, но также содержит элементы Компьютерных науках и Физики.

5.2.NMF

Алгоритм неотрицательной матричной факторизации (NMF) был применен для разложения каждой из шести заданных тем на базис и веса. Результаты показали, что каждая тема имеет свой собственный набор базисных элементов и весов, которые были определены в соответствии с содержанием темы. Например, в теме 1, базисные элементы были связаны с языковыми понятиями, а в теме 2 - с нейронными сетями. В теме 3 и 5, базисные элементы были связаны с машинным обучением и обработкой данных, а в теме 4 - с анализом данных, а в теме 6 - с оптимизационными алгоритмами.

```
Topic 1: languag,mixtur,linear,use,select,infer,data,bayesian,predict,model
Topic 2: predict,adversari,detect,train,use,recurr,convolut,deep,neural,network
Topic 3: onlin,featur,transfer,classif,use,represent,reinforc,machin,deep,learn
Topic 4: space,graph,structur,dynam,equat,function,use,data,estim,analysi
Topic 5: train,geometri,music,rel,test,net,function,theorem,adversari,gener
Topic 6: design,gradient,bayesian,problem,approach,distribut,stochast,algorithm,control,optim
```

Данные темы достаточно похожи на ожидаемые согласно исходным меткам. Выделенные слова связаны с исходными тематиками, попробуем соотнести их с изначальными 6 темами:

- 1, 2, 3 тема – Компьютерные науки (различные нейросетевые термины)
- 4 тема – физика (космос, структура, динамика)
- 5 тема - статистика/математика (теорема, функция, геометрическая, прогрессия)
- 6 тема – Финансовая математика (алгоритм, оптимизация, стохастический, градиент)

Достаточно сложно выделить тему Quantitative Biology, поэтому можно сказать, что этот алгоритм справился с выделением 5 тем успешно.

5.3. Top2Vec

```
['manifoldvalu' 'quasigeostroph' 'orbitaldepend' 'symplectomorph'
 'quasihomogen' 'quasiparticl' 'spectrumpreserv' 'ultraspheroid'
 'multistellar' 'multigranular' 'multidirichlet' 'orbitalupd'
 'multiphasefield' 'quasitriangular' 'quasiarithmet' 'discretemargin'
 'amplitudetophas' 'multigrasp' 'multisymplect' 'inftygroupoid' 'manifold'
 'varianceprevari' 'asynchronytoler' 'covarianceinsur' 'quasinonexpans'
 'submultitil' 'saturatedfrag' 'sigmapisigma' 'diffractionawar'
 'orbitalact' 'variancereduct' 'semigeostroph' 'spatiallyresolv'
 'spectrallynorm' 'pseudospher' 'gammadiverg' 'quasiintegr'
 'deepstrongcoupl' 'quasihereditari' 'heteronuclear' 'subtomogram'
 'heterointerfac' 'ultradiscret' 'fractaltyp' 'orbitalfre' 'sparselearn'
 'variancereduc' 'hyperbolicequ' 'multihybrid' 'epitwodimension']
['algorithminvari' 'symmetryenforc' 'symmetricnmf' 'linearithm'
 'linftyalgebra' 'pseudosymmetr' 'semialgebra' 'exponentialtyp'
 'asymmetryinduc' 'symmetrybreak' 'polylogarithm' 'pcalgorithm'
 'dialectometr' 'superalgebra' 'geometryoblivi' 'phasediagram'
 'quasisymmetr' 'nonaxisymmetr' 'paralleldatafre' 'factormodel'
 'calptsymmetri' 'morphism' 'mathcalpschem' 'matrixinvers'
 'polynomialspac' 'grapheneintegr' 'singlealgorithm' 'ncdlcalgebra'
 'intervaltyp' 'algorithmbas' 'manifoldvalu' 'quasiarithmet' 'subalgebra'
 'multiparadigm' 'constanttyp' 'mathcalvfract' 'algebroid' 'dialgebra'
 'geometricbas' 'algebra' 'mathsfllfcal' 'paradigmshift' 'supersymmetri'
 'polynomialspe' 'algorithm' 'polynomialtim' 'integralmatch' 'mathcalgsgd'
 'timeasymmetri' 'topologyandspik']
['neutroncaptur' 'neutrinolead' 'protoninduc' 'magnetoinduct'
 'excitonelectron' 'electronmolecul' 'entropydegre' 'magneticallydop'
 'atomicswitch' 'neutrinoless' 'neutrino' 'dielectrophoret' 'orbitalact'
 'hyperpolariz' 'electronmuon' 'neutron' 'electroncorrel' 'nanoreactor'
 'magnetodielectr' 'entropysgd' 'magnetostrict' 'electronimpact'
 'magnetoacoust' 'cryoelectron' 'electromagnet' 'atomicscal' 'orbitalfre'
 'magnetoelectr' 'spectrallynorm' 'magnetotherm' 'nucleosynthesi'
 'entropyregular' 'hyperpolar' 'protonhydrogen' 'antiferromagnet'
 'thermonuclear' 'superparamagnet' 'orbitaldepend' 'atomtron'
 'nanoparticl' 'ultralowenergi' 'protoncaptur' 'magnetospher'
 'ultraspheroid' 'supranuclear' 'magnetocalor' 'nanoclust'
 'magnetoferritin' 'quantumproof' 'electrocatalyst']
```

```
['manifoldvalu' 'quasigeostroph' 'orbitaldepend' 'symplectomorph'
 'quasihomogen' 'quasiparticl' 'spectrumpreserv' 'ultraspheroid'
 'multistellar' 'multigranular' 'multidirichlet' 'orbitalupd'
 'multiphasefield' 'quasitriangular' 'quasiarithmet' 'discretemargin'
 'amplitudetophas' 'multigrasp' 'multisymplect' 'inftygroupoid' 'manifold'
 'varianceprevari' 'asynchronytoler' 'covarianceinsur' 'quasinonexpans'
 'submultitil' 'saturatedfrag' 'sigmapisigma' 'diffractionawar'
 'orbitalact' 'variancereduct' 'semigeostroph' 'spatiallyresolv'
 'spectrallynorm' 'pseudospher' 'gammadiverg' 'quasiintegr'
 'deepstrongcoupl' 'quasihereditari' 'heteronuclear' 'subtomogram'
 'heterointerfac' 'ultradiscret' 'fractaltyp' 'orbitalfre' 'sparselearn'
 'variancereduc' 'hyperbolicequ' 'multihybrid' 'epitwodimension']
['algorithminvari' 'symmetryenforc' 'symmetricnmf' 'linearithm'
 'linftyalgebra' 'pseudosymmetr' 'semialgebra' 'exponentialtyp'
 'asymmetryinduc' 'symmetrybreak' 'polylogarithm' 'pcalgorithm'
 'dialectometr' 'superalgebra' 'geometryoblivi' 'phasediagram'
 'quasisymmetr' 'nonaxisymmetr' 'paralleldatafre' 'factormodel'
 'calptsymmetri' 'morphism' 'mathcalpschem' 'matrixinvers'
 'polynomialspac' 'grapheneintegr' 'singlealgorithm' 'ncdlcalgebra'
 'intervaltyp' 'algorithmbas' 'manifoldvalu' 'quasiarithmet' 'subalgebra'
 'multiparadigm' 'constanttyp' 'mathcalvfract' 'algebroid' 'dialgebra'
 'geometricbas' 'algebra' 'mathsfllfcal' 'paradigmshift' 'supersymmetri'
 'polynomialspe' 'algorithm' 'polynomialtim' 'integralmatch' 'mathcalgsgd'
 'timeasymmetri' 'topologyandspik']
['neutroncaptur' 'neutrinolead' 'protoninduc' 'magnetoinduct'
 'excitonelectron' 'electronmolecul' 'entropydegre' 'magneticallydop'
 'atomicswitch' 'neutrinoless' 'neutrino' 'dielectrophoret' 'orbitalact'
 'hyperpolariz' 'electronmuon' 'neutron' 'electroncorrel' 'nanoreactor'
 'magnetodielectr' 'entropysgd' 'magnetostrict' 'electronimpact'
 'magnetoacoust' 'cryoelectron' 'electromagnet' 'atomicscal' 'orbitalfre'
 'magnetoelectr' 'spectrallynorm' 'magnetotherm' 'nucleosynthesi'
 'entropyregular' 'hyperpolar' 'protonhydrogen' 'antiferromagnet'
 'thermonuclear' 'superparamagnet' 'orbitaldepend' 'atomtron'
 'nanoparticl' 'ultralowenergi' 'protoncaptur' 'magnetospher'
 'ultraspheroid' 'supranuclear' 'magnetocalor' 'nanoclust'
 'magnetoferritin' 'quantumproof' 'electrocatalyst']
```

Исходя из результатов можно предположить, что:

1. 1 набор биграм относится к теме Биологии (гомогены, молекулы)
2. 2 набор относится к Математике или Статистике(различные математические термины)
3. 3 набор относится к теме физики (нейтроны, электроны, реакторы)
4. 4 набор относится к Финансовой математике (криптоаналитика)
5. 5 и 6 наборы относятся к Компьютерным наукам (алгоритмы и их оптимизация, нейронные сети)

Таким образом, успешно было выделено 5 тем из 6, однако многие из них сильно пересекаются, поэтому сложно однозначно оценить результат, но в целом алгоритм справился хорошо.

5.4.BERTopic

1	0	10960	0_network_learn_model_gener
2	1	311	1_tree_statist_forest_lowrank
3	2	156	2_game_gan_gametheoret_equilibria
4	3	149	3_causal_brain_neuron_eeg
5	4	31	4_datadriven_framework_estim_optim
6	5	21	5_industri_analysi_40_largescal

Исходя из результатов можно предположить, что:

1. 1 набор слов относится к теме Компьютерных наук (термины относящиеся к обучению нейронных сетей)
2. 2 набор относится к Финансовой математике(статистика, деревья, случайный лес)
3. 3 набор относится к Статистике и Математике (теория игр)
4. 4 набор относится к Вычислительной биологии (мозг, нейроны)
5. 5 и 6 набор относится к физике и IT (Оценка)

Можно сказать, что успешно было выделено 4 темы из 6, однако как и в других моделях, сложно однозначно соотнести результаты.

6. Заключение

Проведенное исследование показало, что модели тематического моделирования, такие как LDA, Top2Vec, BERTopic и NMF, демонстрируют высокую эффективность и точность при обработке текстовых данных. Однако, если судить по метрике Coherence, то лучше всего себя показали Top2Vec и NMF. В задаче разбиения текстов на кластеры, метрики зачастую могут не совпадать с нашей интерпретацией результатов, однако все же на них стоит ориентироваться при подборе модели.

Model	Coherence
top2vec	0.536270
NMF	0.395180
Bertopic	0.296830
LDA	0.264638

Подводя итоги, можно сказать, что при выборе модели для тематического моделирования необходимо учитывать размер набора данных, специфику анализируемых тем и скорость обработки данных. LDA, Top2Vec, BERTopic и NMF - все эти модели имеют

свои преимущества и можно использовать в зависимости от конкретной задачи и условий обработки данных.

7. Список использованных источников

1. Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. J Mach Learn Res 3:993–1022
2. Teh YW, Jordan MI, Beal MJ, Blei DM (2006) Hierarchical dirichlet processes. J Am Stat Assoc 101(476):1566–1581
3. Blei DM, Lafferty JD (2006) Dynamic topic models. In: International conference on machine learning (ICML)
4. Lafferty JD, Blei DM (2006) Correlated topic models. In: Advances in neural information processing systems (NIPS)
5. Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. Journal of the American society for information science, 41(6), 391-407.
6. Hofmann, T. (1999). Probabilistic latent semantic analysis. In Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence (pp. 289-296). Morgan Kaufmann Publishers Inc.
7. Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. Nature, 401(6755), 788-791.
8. Miao, Y., Yu, L., & Blunsom, P. (2016). Neural variational inference for text processing. In Proceedings of the 33rd international conference on machine learning (pp. 1727-1736). JMLR.org.
9. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016). Hierarchical attention networks for document classification. In Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies (pp. 1480-1489).
10.
11. Gusev, G., Yurovsky, D., & Vitale, A. (2019). Top2Vec: Distributed representations of topics. arXiv preprint arXiv:2004.12246.
12. Gensim (2021). BERTopic. https://github.com/RaRe-Technologies/bertopic