Тематическое моделирование научных статей

Бондаренко Даниил

11.03.2023

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 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.

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.

Результаты работы

В данной работе был использован датасет, состоящий из названий научных статей по 6 разным направлениям:

- Информационные технологии
- Физика
- Математика
- Статистика
- Биотех
- Финтех

Для данного датасета лучше всего показали себя такие подходы как LDA и NMF. В свою очередь, они являлись самыми быстрыми по времени исполнения. Можно предположить, что в данном случае лучше всего показали себя эти подходы, так как в данных отсутствовали сложные зависимости, а такие подходы как Top2Vec и BERTopic могут лучше себя показывать на более сложных контекстных данных.

LDA

```
[(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"')]
```

NMF

```
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
```

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']
```

```
['wholedetector' 'cheateridentifi' 'detector' 'multisensori'
 'pseudoproven' 'detect' 'selfmonitor' 'multisensor' 'sensorimotor'
 'signaldetect' 'spectrogram' 'perceptionact' 'photodetector'
 'hyperspectr' 'tracenorm' 'intuitionist' 'radarcommun' 'observationbas'
 'selectivespectr' 'neuralprosthet' 'spectrumpreserv' 'pseudospectr'
 'perceptionbas' 'subtomogram' 'betweenstudi' 'cryptanalysi'
 'neuroimagingbas' 'selfindex' 'neuralnetwork' 'oversight' 'metastudi'
 'photodetect' 'selflearn' 'probe' 'spectrallead' 'simulationoptim'
 'discern' 'trackingfilt' 'dialectometr' 'interferometri' 'machinelearn'
 'encoderdecod' 'nearsensor' 'neuromodul' 'visuallinguist'
 'algorithminvari' 'surveyplanet' 'runandinspect' 'multiperspect'
 'observatori']
['optimallyplac' 'optimizationbas' 'optimalr' 'optimizatio'
 'algorithminvari' 'optimparallel' 'accuracyeffici' 'arbitraryprecis'
 'pcalgorithm' 'quasidetermin' 'bestmatch' 'variancestabil' 'algorithm'
 'superalgebra' 'biascorrect' 'algorithmbas' 'quasiarithmet'
 'singlealgorithm' 'optimis' 'exponentialtyp' 'strictlycorrel' 'approach'
 'quasiperfect' 'integralmatch' 'perfectoid' 'pseudorandom'
 'moderateconfid' 'optimist' 'randomcoeffici' 'qloglikelihood'
 'paradigmshift' 'modestlyinclin' 'symmetrybreak' 'randomnessinduc'
 'quasipolynomi' 'variancereduct' 'losslikelihood' 'radiancepredict'
 'delayconstrain' 'pragmat' 'pseudoproven' 'likelihoodfre'
 'quasigeostroph' 'quasiconform' 'quasiregular' 'quasiintegr'
 'superexponenti' 'derivativefre' 'probabilist' 'variancereduc']
['neuralnetwork' 'networkscal' 'networktheoret' 'networkanalyt'
 'networkstructur' 'network' 'trajectorynet' 'networklevel' 'networksbas'
 'subnetwork' 'neuroimagingbas' 'intranet' 'networkbas' 'hybridnet'
 'neuralprosthet' 'inversefacenet' 'clusternet' 'neuralbran' 'neuralguid'
 'neuralpow' 'neuroninspir' 'neuromodul' 'deepnet' 'neural' 'neuromorph'
 'cyberwarfar' 'neuron' 'rede' 'evflownet' 'conceptnet' 'neuroimag'
 'neuralbas' 'pulsewidth' 'neuromechan' 'neurodegen' 'bandwidth'
 'neurorul' 'scatternet' 'cyberattack' 'neurodynam' 'cyberphys'
 'neuroanatom' 'neurofeedback' 'neuromuscular' 'bitwidth' 'broadband'
 'qneuron' 'neurofuzzi' 'wavenet' 'pathwidth']
```

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

Список использованной литературы

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