Scientific Research Publishing

Open Journal of Applied Sciences, 2021, 11, 541-548
https://www.scirp.org/journal/ojapps
ISSN Online: 2165-3925
ISSN Print: 2165-3917

Identification of Topics from Scientific Papers through Topic Modeling

Denis Luiz Marcello Owa

Pontifical Catholic University of Sao Paulo, Sao Paulo, Brazil
Email: denis0118@yandex.com

How to cite this paper: Owa, D.L.M. (2021) Identification of Topics from Scientific Papers through Topic Modeling. *Open Journal of Applied Sciences*, **11**, 541-548.
https://doi.org/10.4236/ojapps.2021.114038

Received: March 19, 2021
Accepted: April 26, 2021
Published: April 29, 2021

Abstract

Topic modeling is a probabilistic model that identifies topics covered in text(s). In this paper, topics were loaded from two implementations of topic modeling, namely, Latent Semantic Indexing (LSI) and Latent Dirichlet Allocation (LDA). This analysis was performed in a corpus of 1000 academic papers written in English, obtained from PLOS ONE website, in the areas of Biology, Medicine, Physics and Social Sciences. The objective is to verify if the four academic fields were represented in the four topics obtained by topic modeling. The four topics obtained from Latent Semantic Indexing (LSI) and Latent Dirichlet Allocation (LDA) did not represent the four academic fields.

Keywords

Topic Modeling, Corpus Linguistics, Gensim, LSI, LDA

1. Introduction

Topic modeling is a relevant technique in Natural Language Processing that can be used to extract information from a text or a group of them to discover their topics. In other words, it is a method that identifies themes or ideas underlying a text or a corpus. Considering that topic modeling is a resource that can be used for summarizing a large number of texts, it is useful for Corpus Linguistics.

According to [1], topic modeling has been used in many areas like sociology, digital humanities, political science, literary studies and academic discourse.

In this study, topic modeling will be performed in a corpus of 1000 scientific papers, written in English, obtained from PLOS ONE website, from four scientific areas (Biology and life sciences, Medicine and health sciences, Physical sciences and Social sciences). The goal is to verify if the generated topics represent these four areas.

2. Related Work

The first study on topic modeling related to scientific papers was [2], that obtained topics using LDA. The authors presented significant aspects of the structure of science that stood out, in addition to revealing some relationships between scientific articles in different disciplines. [3] presented a study on accessibility and trust regarding topic modeling. The authors used the discontinued Stanford Topic Modeling Toolbox software on a corpus of text from a social networking. [4] presented the Topic-Aspect Model (TAM) to model a corpus of computational linguistics abstracts and find that the scientific topics identified in the data tend to include both a computational aspect and a linguistic aspect. [5] proposed an algorithm for recommending scientific articles to users based on both content and other users' ratings. Their study presented an approach that works well relative to traditional matrix factorization methods and made good predictions on completely unrated articles. [1] demonstrated the use of topic models to explore a corpus of specialized English discourse. The authors employed topic models to investigate within-paper topical change, examine the chronological change of a journal, identify different types of papers, and differentiate multiple senses of words. [6] presented additive regularization when creating models to highlight the groups of topics in the Probabilistic Latent Semantic Analysis.

3. Theoretical Framework

Topic modeling is a technique to obtain, using probabilistic calculations, which topics are represented in a text or a group of texts. Topics are a group of words that load together after those calculations. A topic can be understood as a theme or an underlying idea represented in a text. For example, if we are working with a corpus of newspaper articles, some possible topics could be weather, politics, sport, and so on [7].

Every text can be classified into a main subject, however other subjects are present in different degrees. According to [8], there are three widely used software for topic modeling: Gensim, Mallet and LDA in R. Stanford Topic Modeling Toolbox was discontinued, as their address (https://nlp.stanford.edu/software/tmt) has been down for months.

Gensim (Generate Similar[1]) is a free open-source Python library that processes plain texts and uses unsupervised[2] machine learning algorithms. It is designed to work with large corpora. Mallet (MAchine Learning for LanguagE Toolkit[3]) is a Java-based package for statistical natural language processing, document classification, clustering, topic modeling, information extraction, and other machine learning applications to text. LDA in R[4] implements latent Dirichlet allocation

[1]https://radimrehurek.com/gensim/index.html.

[2]In unsupervised learning, there is no distinction between training and test data. The learner processes input data with the goal of creating some summary, or compressed version of that data [9].

[3]http://mallet.cs.umass.edu/topics.php.

[4]https://cran.r-project.org/web/packages/lda.

(LDA) and related models, such as sLDA, corrLDA, and the mixed-membership stochastic blockmodel.

Gensim was used for two reasons: 1) the author is familiar with Python coding; 2) it is possible to use a wrapper in Gensim for Latent Dirichlet Allocation (LDA) from Mallet, but the opposite is not possible (to use Gensim in Mallet).

Gensim implemented the topic modeling central concepts as explained by [10]:

1) Document: any text (a string, or *str* in Python). A document can be a tweet, a single paragraph, a summary of a newspaper article, an entire article or a book.

2) Corpus: collection of Document objects. The class gensim.corpora.Dictionary creates a dictionary with the words in the corpus, each word having a unique identification number (or ID).

3) Vector: a mathematical representation of the texts. A widely used vector is the bag-of-words model. This model consists of creating a representation for each document with frequency counts for each word in the dictionary. Under the bag-of-words model, each document is represented by a Vector containing the frequency counts of each word in the dictionary. One of the main properties of the bag-of-words model is that it completely ignores the order of the tokens in the encoded document, which is why it is called bag-of-words.

4) Model: transformation from one vector to another. Gensim offers five models: Latent Semantic Indexing (LSI), Latent Dirichlet Allocation (LDA), Tf-Idf, Random Projections (RP) and Hierarchical Dirichlet Process (HDP). They are explained below:

a) Latent semantic indexing (LSI): According to [11], LSI is based on linear algebra. Its information retrieval technique is based on the spectral analysis of the term-document matrix. LSI can capture the underlying semantics of the corpus and achieve improved retrieval performance. According to [12], the characteristics derived from LSI, which are linear combinations of the original Tf-Idf characteristics, can capture some aspects of basic linguistic notions, such as synonymy and polysemy. [13] states that LDA was developed to fix an issue with the previously developed probabilistic latent semantic analysis.

b) Latent Dirichlet allocation (LDA): this is most frequently used implementation for topic modeling recently. According to [12], the LDA is a generative probabilistic model for collections of discrete data, such as text corpora. LDA is a three-level, hierarchical Bayesian model, in which each item in a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. [13] states that a distinguishing characteristic of LDA is that all documents in the collection share the same set of topics, but each document exhibits those topics in different proportions.

c) Term Frequency * Inverse Document Frequency (Tf-Idf): According to [12], this model generates a basic vocabulary of words or terms and, for each document in the corpus, a count of the number of occurrences of each word is

formed. After proper normalization, this term frequency count is compared to an inverse document frequency count, which measures the number of occurrences of a word throughout the corpus (on a logarithmic scale and again properly normalized). The final result is a term-by-document X matrix, whose columns contain the Tf-Idf values for each document in the corpus.

d) RP: aims to reduce the dimensionality of the vector space. RP is an efficient approach (friendly to memory and CPU) to approximate Tf-Idf distances between documents, adding a little randomness, as explained by [14].

e) Hierarchical Dirichlet process (HDP): According to [15], HDP is a non-parametric Bayesian model that can be used for model-mixed association data with a potentially infinite number of components. It has been largely used in probabilistic topic modeling, where data are documents and the components are distributions of terms that reflect recurring patterns (or topics) in the collection. A limitation of HDP analysis is that existing inference algorithms require multiple passages through all data, *i.e.*, these algorithms are intractable for large-scale applications. Eventually, the number of topics is determined by the model itself.

In this paper, the author will work with LSI and LDA algorithms for two reasons: 1) it is possible to determine in advance the number of topics to be found; and 2) LDA is an improvement from LSI, so the author is investigating the results between the two algorithms.

4. Methodology

The corpus from this study has 1000 scientific papers, published in English, obtained from the PLOS ONE website (https://journals.plos.org/plosone). The author named this corpus as Corpus of Four Fields from PLOS ONE (CFFPO).

PLOS ONE is an open-access, peer-reviewed scientific journal published by the Public Library of Science (PLOS) since 2006. In April 2021, the site has 249,849 published papers, subdivided into the following areas:

1) Biology and life sciences
2) Computer and information sciences
3) Earth sciences
4) Ecology and environmental sciences
5) Engineering and technology
6) Medicine and health sciences
7) People and places
8) Physical sciences
9) Research and analysis methods
10) Science policy
11) Social sciences

For this study, 250 papers from biology and life sciences, Medicine and health sciences, Physical sciences and Social sciences were downloaded. Each paper has five parts, namely: 1) abstract; 2) introduction; 3) methods; 4) results and discussion; 5) conclusions.

Table 1 below shows the design of CFFPO, according to LancsBox software[5]:

These texts were downloaded through an internet browser from the address http://api.plos.org/search, using parameters on the URL to bring papers from each area.

For example, the address below retrieves 250 papers from Biology and life sciences, containing the paper ID (for example, 10.1371/journal.pbio.2006735), the subject and parts introduction, abstract, conclusions, materials and results:

http://api.plos.org/search?q=subject:%22Biology%20and%20life%20sciences22+and+doc_type:full&fl=id,subject,introduction,abstract,conclusions,materials_and_methods,results_and_discussion&start=1&rows=250

The 1000 papers are initially retrieved in *.json format[6]. This format is not readable by Gensim, which accepts other formats, like a raw text (*.txt). A Python program was run to parse from json to txt.

The corpus was uploaded to Google Drive, to be able to execute the topic modeling program in Google Colaboratory. Collaboratory, or "Colab", is a cloud environment for programming in which Python programs can be written and executed directly from the internet browser, without the need to configure the computer itself. Some advantages are the following: 1) access to an environment configured to run programs; 2) access to the computational resources of Google's servers; 3) ease of sharing codes and results. Colab eventually offered a good environment for Gensim codes, because running topic modeling codes over big corpora takes hours when run in ordinary laptops.

The Python code that performs topic modeling for CFFPO consists of the following algorithm:

1) The entire CFFPO content is loaded into the computer's RAM.

2) For each text in the corpus, its content is transformed into lowercase tokens and put into a list (a data type of the Python language);

3) SpaCy parser performs part-of-speech tagging;

4) Named entities[7] are excluded;

5) Only nouns, adjectives, adverbs and verbs are kept;

Table 1. Design of the CFFPO.

Scientific field	Texts	Tokens	Types	Lemmas
Biology and life sciences	250	1.514.210	92.957	86.077
Medicine and health sciences	250	992.818	81.399	74.090
Physical sciences	250	1.323.422	78.670	71.722
Social sciences	250	1.532.088	89.321	82.198
Total	**1.000**	**5.362.538**	**342.347**	**314.087**

Source: LancsBox.

[5]http://corpora.lancs.ac.uk/lancsbox.

[6]JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript (https://www.json.org/json-en.html).

[7]Roughly speaking, anything that can be referred to with a named entity proper name: a person, a location, an organization [16].

6) Stopwords[8] are removed

7) The lemma of the words is obtained;

8) Bigrams (consecutive two-word sequence) are built. Trigrams would take long hours to process, so they are not used here;

9) A corpus bag-of-words is prepared, one for each text;

10) This bag-of-words is converted into tuples (a type of immutable word list in Python). Each document is therefore converted to a tuple;

11) LSI and LDA implementations are applied to these tuples.

The author configured LSI and LDA implementations to identify four topics each. The reason is to verify if the four academic areas of CFFPO are represented through topic modeling.

5. Results

The topics loaded with LSI implementation are the following:

Topic 1: study, fig, high, result, table, show, gene, include, patient, group

Topic 2: patient, gene, study, protein, fig, year, expression, plant, group, risk

Topic 3: patient, gene, model, protein, expression, value, plant, treatment, measure

Topic 4: patient, gene, fig, concentration, participant, protein, increase, population, water, individual

The topics loaded with LDA implementation are the following:

Topic 1: fig, gene, protein, show, high, result, increase, table, sample, value

Topic 2: study, model, result, high, level, participant, group, number, measure, include

Topic 3: population, fig, table, number, snps, study, base, analysis, result, high

Topic 4: patient, study, year, group, include, risk, high, treatment, table, follow

6. Discussion

LSI implementation loaded four topics related to Medicine and Biology, but none of them specifically belong to a scientific field. The topics loaded with LSI excluded Physics nor Social Sciences.

With LDA implementation, there are some differences. Topic 1 contains words typical of biological studies, such as "gene" "protein" and "sample" and is therefore adherent to Biology. Topic 2 loaded words used in all four fields of CFFPO. These words compose a generic topic related to all those four fields. A search for the words from topic 2 within the content of the files returned papers from the four scientific fields. It confirms that this topic is generic to all four fields. Topic 3 is related to Biology, with the word snps (Single nucleotide polymorphisms) loaded into this topic. Other words such as "population", "table", "number" and "study" in fact appear within the Biology papers of CFFPO. Finally, topic 4 is a topic from Medicine papers. It is possible to observe, therefore, that the scientific fields of Physics and Social Sciences were not represented in

[8]List of most common words of a language useful to filter out, for example "and" or "I".

any topic loaded by LDA.

Some explanations are possible. First, the papers published in the areas of Biology and Medicine share many technical words to describe their methods, results and discussions. Second, the scientific papers in Physics and Social Sciences of CFFPO are not about themes using recurring words, *i.e.*, each paper in Physics and Social Sciences were about different subjects from each other. Possibly, those papers did not use recurring words. Considering that topic modeling is a statistical tool, based on recurring words within documents, the topics loaded in this study were unable to identify a topic from these scientific areas.

LDA loaded a generic topic (topic 2) in the results. Its words are used in all four areas. However, LDA did not load any specific topic from Physics or Social Sciences papers. LSI loaded topics from Biology and Medicine and did not load a generic topic about three or more areas.

7. Future Research

Comparing topics loaded by Gensim and by Mallet on the same corpus will bring interesting results. Here, only LDA algorithm is possible. The identification of topics exclusively from a specific scientific area, and not all areas together, will refine the topics to each area.

8. Conclusion

This study presented topics generated by topic modeling on a corpus of 1000 scientific papers in English, from the areas of Biology and life sciences, Medicine and health sciences, Physical sciences and Social sciences. The algorithms LSI and LDA were used to identify four topics from each area to investigate if the four major academic areas were represented. This expected result was not accomplished. The four topics from LSI were related to the areas of Biology and Medicine, and the four topics from LDA had two topics related to Biology, one related to Medicine and one generic topic. One possible reason for that is that Biology and Medicine share many technical words to describe their methods, results and discussions. Physical sciences and Social sciences were not represented among the topics loaded from any algorithm.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Murakami, A., Thompson, P., Hunston, S. and Vajn, D. (2017) What Is This Corpus about? Using Topic Modelling to Explore a Specialised Corpus. *Corpora*, **12**, 243-277. https://doi.org/10.3366/cor.2017.0118

[2] Griffiths, T.L. and Steyvers, M. (2004) Finding Scientific Topics. *Proceedings of the National Academy of Sciences of the United States of America*, **101**, 5228-5235. https://doi.org/10.1073/pnas.0307752101

[3] Ramage, D., Rosen, E., Chuang J., Manning, C. D. and McFarland, D. A. (2009) Topic Modeling for the Social Sciences. NIPS-Workshop on Applications for Topic Models: Text and Beyond. https://nlp.stanford.edu/dramage/papers/tmt-nips09.pdf

[4] Paul, M. and Girju, R. (2010) A Two-Dimensional Topic-Aspect Model for Discovering Multi-Faceted Topics. *Proceedings of the National Conference on Artificial Intelligence*, Atlanta, 11-15 July 2010, 1.

[5] Wang, C. and Blei, D.M. (2011) Collaborative Topic Modeling for Recommending Scientific Articles. *Proceedings of the* 17*th ACM*|*SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, August 2011, 448-456. https://doi.org/10.1145/2020408.2020480

[6] Krasnov, F. (2018) Topic Classification through Topic Modeling with Additive Regularization for Collection of Scientific Papers. *Proceedings of the* 14*th Central and Eastern European Software Engineering Conference Russia*, Moscow, October 2018, Article No. 5. https://doi.org/10.1145/3290621.3290629

[7] Srinivasa-Desikan, B. (2018) Natural Language Processing and Computational Linguistics: A Practical Guide to Text Analysis with Python, Gensim, SpaCy and Keras. Packt Publishing Ltd., Birmingham.

[8] Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., *et al.* (2019) Latent Dirichlet Allocation (LDA) and Topic Modeling: Models, Applications, a Survey. *Multimedia Tools and Applications*, **78**, 15169-15211. https://doi.org/10.1007/s11042-018-6894-4

[9] Shalev-shwartz, S. and Ben-david, S. (2014) Understanding Machine Learning from Theory to Algorithms. Cambridge University Press, Cambridge, 4-5.

[10] Řehůřek, R. (2020) Core Concepts. https://radimrehurek.com/gensim/auto_examples/core/run_core_concepts.html#what-is-gensim

[11] Papadimitriou, C.H., Raghavan, P., Tamaki, H. and Vempala, S. (2000) Latent Semantic Indexing: A Probabilistic Analysis. *Journal of Computer and System Sciences*, **61**, 217-235. https://doi.org/10.1006/jcss.2000.1711

[12] Blei, D., NG, A.Y. and Jordan, M.I. (2003) Latent Dirichlet Allocation. *The Journal of Machine Learning Research*, **3**, 993-1022.

[13] Blei, D. (2012) Probabilistic Topic Models. *Communications of the ACM*, **55**, 77-84. https://doi.org/10.1145/2133806.2133826

[14] Řehůřek, R. (2020) Topics and Transformations. https://radimrehurek.com/gensim/auto_examples/core/run_topics_and_transformations.html#sphx-glr-auto-examples-core-run-topics-and-transformations-py

[15] Wang, C., Paisley, J. and Blei, D. (2011) Online Variational Inference for the Hierarchical Dirichlet Process. *Journal of Machine Learning Research-Proceedings Track*, **15**, 752-760.

[16] Jurafsky, D. and Martin, J.H. (2019) Speech and Language Processing. https://web.stanford.edu/~jurafsky/slp3/old_oct19/17.pdf