*International Journal of Innovative Computing, Information and Control*
Volume **17**, Number **5**, October **2021**

ICIC International ©2021 ISSN 1349-4198
pp. **1643–1651**

# AUTOMATIZATION NEWS GROUPING USING LATENT DIRICHLET ALLOCATION FOR IMPROVING EFFICIENCY

Tirza Madah Pratidina and Djoko Budiyanto Setyohadi

Informatics Department
Universitas Atma Jaya Yogyakarta
No. 43, Babarsari Street, Yogyakarta 55281, Indonesia
205303222@students.uajy.ac.id; djoko.budiyanto@uajy.ac.id

Received December 2020; revised May 2021

Abstract. *News is one of the most frequently occurring text data in human life. The news appears in the form of printed news or online news. With new news every day, there is a lot of news in various life fields. With that different kind of news, the redactor needs to read them and then divide them into various fields. To group news into their respective fields, people need a lot of time to read the news and group them into the appropriate fields. With this condition, soon the tasks will be piled up. News grouping will also help the reader to find news based on the topic that they want to read. The existing problems inspire the researcher to conduct research about grouping news automatically. This research will use the Latent Dirichlet Allocation (LDA) method with perplexity calculation for choosing the number of the topics that gives a better prediction. The optimal number of topics that found after a comparison of alpha value and perplexity value is twenty-nine topics with a perplexity value of 997.5468 and an alpha value of 0.2. The result of this research is groups of the word that can be concluded into many topics.*
**Keywords:** Topic modelling, News, Latent Dirichlet allocation

1. **Introduction.** Every day various events occur around the world. In the journalism field, they will transform them into some news and publish it to the public [1]. The news appears in the form of printed news or online news. Online news is often published everyday anytime. These news articles create new perceptions about something and influence readers in their daily lives [2]. With new news every day, there is a lot of news in various life fields, such as sports, culinary, education, and entertainment news. With that different kind of news, the redactor needs to read them and then divide them into various fields.

The increasing number of readers in text online, especially news, is in line with the increasing use of NLP (Natural Language Processing) and machine learning to perform automated analysis of news texts. This technology enables humans to obtain information from large amounts of data, especially in the form of text, to support the process of producing news and up to content analysis. One example of NLP in the context of news text analysis is the discovery of a topic with an unsupervised concept from a large set of texts [3]. The increasing number of online news in the form of text led to the emergence of the idea to conduct news text analysis [4]. To group news into their respective fields, of course, people who share them need a lot of time to read the news and group them into the appropriate fields [5]. Manually do this work with limited processing power, resulting in a few numbers of news analyzed. Meanwhile, the news is always updated every time or every day in a short time; it will make the works piled up. The existing problems inspire

DOI: 10.24507/ijicic.17.05.1643

researchers to conduct research that can shorten the time needed to classify existing news. This research focused on shortening the time used to divide a lot of news into its relevant field compared to doing this task manually.

In this study, we used topic modeling to conduct topics based on the news that we analyzed. Topic modeling is one interesting topic since it can be used to explore the important structure within document. The common approach such as Latent Semantic Analysis (LSA) and Probabilistic Latent Semantic Analysis (pLSA) has potential problem especially on overfitting problem when they deal with topic analysis. *Dirichlet* as main tools in Latent Dirichlet Allocation (LDA) is aimed to cop the problem. Furthermore, it can be used for expanding pLSA by mixing related parameter on *Dirichlet* [6].

This new method of grouping news will be of great use to the redactors and the reviewers of news in their organization. People who work in a television station, especially those who work as news sorters and news categorizers also need this new way of doing news grouping. In this paper, we compare the perplexity value to determine the optimal number of topics. What makes the difference from the previous paper is that the writer compares the perplexity value of different alpha with a value range of 0 to 1. From this comparison, the lowest perplexity value of each alpha will be compared from the smallest value sought. Then, group the news into some areas resulting from the analysis process. This paper is divided into several parts. In Section 1, we explain why this research is important and the main purpose of this research. In Section 2, we do an overview of the previous works related to this paper, explain the method used in this paper and how our data can be calculated with that method. In Section 3, we explain the result of the LDA method and the perplexity measures. In Section 4, we analyze and evaluate these results and compare them against the previous work that is related to this research. In Section 5, we conclude the paper and analyze what the next researcher can do in future work.

2. **Methods.** Previously, we have also applied the topic modelling method to analyzing text data originating from tweets from hotel accounts. In our research, it aims to find out what things these hotels share on their timeline on Twitter. With the same method, Annisa et al. also did research about the hotel's review. The result of this research is a group of words that are categorized as a topic. Coherence calculation was used to determine the number of the topics that will be searched [7]. Jacobi et al. in 2015 [8] conducted a quantitative analysis of a set of journalistic texts using the topic modelling method. The data used in this research is obtained from a New York Times article on nuclear technology. In their research, Jacobi et al. used latent Dirichlet analysis, which is an unsupervised technique of topic modelling. For the accuracy check, this research uses coherence calculation like what Annisa et al. [7] did in their research. Loureiro et al. in 2018 analyzed journals related to VR and marketing to do topic modelling with the data. Furthermore, the calculation of the number of topics to be searched for and the searching for these topics is using topic modelling using the Gibbs sampling method [9]. Evans in 2014 conducted a computational technique to perform qualitative analysis for sizeable textual data. The data used are from American newspapers from 1980 to 2012. The method used in the study is topic modelling by applying Latent Dirichlet Allocation (LDA). After analysis with LDA, word distribution was generated based on the topic and then labelled each topic manually [10].

The data analyzed in this paper is news data from many news websites. The steps for this research can be seen in Figure 1, and they are: 1) the data will be crawled and saved into the document, 2) then, the pre-processing process will be carried out on the data, 3) then, the data that already clean will be processed to check the optimal number of topics that will be formed, we use perplexity in this step, 4) then we do a comparison of

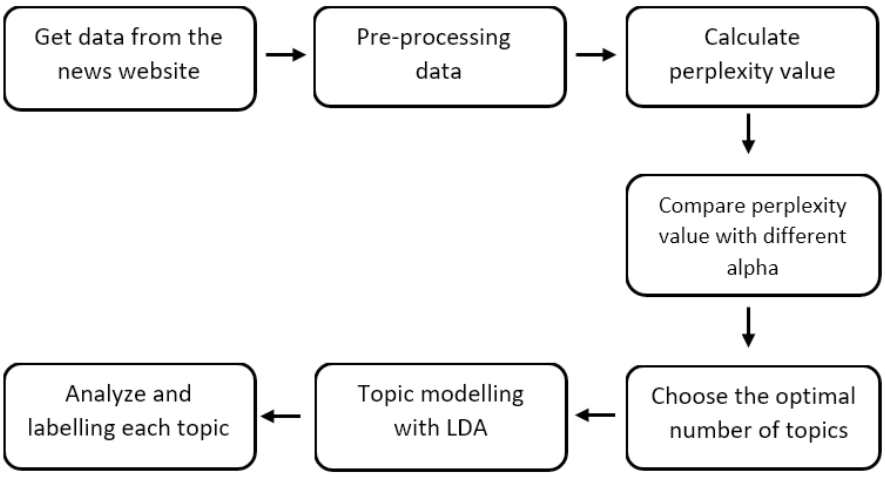


FIGURE 1. Methodology

perplexity value with different alpha that is used, 5) after the comparison, we decide the number of topics that will be used, 6) then we do the analysis process with topic modelling method, 7) and after that, the formed topic can be labelled based on the related words by distributing questionnaires to respondents. In this research, topic modelling was applied using the Latent Dirichlet Allocation (LDA) method. A topic modelling algorithm is a statistical method that analyzes the original text words to find themes or topics in the text [11]. The result of topic modeling method is a group of word that is related to each other in that topic [12]. A piece of news can contain more than one topic. This is similar to the definition of overlapping communities that is explained by Ding et al. [13], who said that overlapping communities happen when one object or node can contain various themes and play a role of several modules. The topic modelling algorithm is an unsupervised method, so it does not require an explanation or labelling of previous documents. Topic modelling allows us to organize and summarize electronic archives at a scale that human annotations would not be possible to do. The purpose of topic modelling is to find topics from a document automatically [14].

The first step is taking data from news sites and then storing it in a file to make it easier to perform data processing. Then proceed with cleaning the existing data or called the pre-processing process. Then the Latent Dirichlet Allocation (LDA) method is used to do topic modelling. The use of topic modelling is to identify a theme or topic in a collection of documents presented in a group of words grouped into several topics [15]. Each word has a different probability of being part of that topic [16]. The most common algorithm to perform topic modelling is Gibbs sampling, and the formula of this algorithm can be seen in Formula (1) [14].

$$P(z_i = j|t_i, d_i, z_{-i}) \propto \frac{C_{t,j}^{(TZ)} + \beta}{\sum_t C_{t,j}^{(TZ)} + T\beta} \frac{C_{d,z}^{(DZ)} + \alpha}{\sum_z C_{d,z}^{(DZ)} + Z\alpha} \tag{1}$$

where $C^{(TZ)}$ is a variable for topic-term assignments, $C^{(DZ)}$ counts document-topic assignments, $z_{-i}$ symbolizes all of the topic-term and document-topic assignments except for $z_i$ term $t_i$, and then $\alpha$ and $\beta$ are *Dirichlet* parameters [17]. To calculate the topic probability of each word, Formula (2) is used where $\varphi_{tz}$ is the probability of the word $t$ for topic $z$.

$$\varphi_{tz} \propto \frac{C_{t,j}^{(TZ)} + \beta}{\sum_t C_{t,j}^{(TZ)} + T\beta} \tag{2}$$

Meanwhile, to calculate the proportion of topic $z$ from each document $d$, Formula (3) can be used

$$\theta_{dz} \propto \frac{C_{d,z}^{(DZ)} + \alpha}{\sum_z C_{d,z}^{(DZ)} + Z\alpha} \tag{3}$$

Before doing this step, we can check the optimal number of the topics that can be generated. Annisa et al. [7] and Jacobi et al. [8] used coherence calculation to find the optimal number of topics. Different from them, we use the perplexity concept to do this task. Perplexity is applied in the steps to evaluating the formed model. Perplexity calculations are often performed in language and text modelling. The result of this calculation is the probability or the likelihood of each word. Perplexity can be defined with Formula (4) [18].

$$perplexity(D_{test}) = \left\{ -\frac{\sum_{d=1}^{M} \log p(w_d)}{\sum_{d=1}^{M} N_d} \right\} \tag{4}$$

In this step, we used our dataset to be calculated with this function. Here, $M$ is the number of documents calculated in this research, $w_d$ is the words contained in document $d$, and $N_d$ is the number of words in document $d$. This activity will repeat the calculation based on our number of iteration that we set before. This formula counts all of the words in the corpus of documents and the result of the calculation can be shown as a graphic. To conclude the result, we can search a number of a topic with a lower score of perplexity. As what Blei et al. said in their paper that a good model could be formed using a number of the topic with a lower perplexity value [18]. After the number of topics found, we can continue to do the analysis process using this LDA method, some topics will be formed, and we will also find news titles related to these topics.

3. **Experiment and Result.** News data that we used in this research are from a trusted news website. We use the one from the BBC News website. To get the data, we used a program based on *Python*. We modify a program from *https://github.com/LuChang-CS/news-crawler* [19]. With that program, we can get the news from the date that we want. After crawling the data from the website, a list of news will be saved on our computer as a .txt file. After we got the data, we can start to do the pre-processing. The file that has been stored is called into the program and then each line of data is converted into characters for easier pre-processing. We used R programming language to perform the pre-processing process. The pre-processing processes are removing links, emojis, removing punctuation marks, removing stopwords, removing numbers, changing letters into lowercases, replacing non-standard words, and also removing affixes. Data cleaning in the form of removing stopwords in the text is using stopwords provided by the *tm* library. However, to make the results more accurate, the researcher also added a list of stopwords obtained from other sources [20]. This pre-processing step is also assisted by the *textclean* library. Then, we calculate the number of topics to be searched with the perplexity calculation. We used a number of a topic with the lowest perplexity value.

In this research, we use the library from R to calculate perplexity. The calculation is done by calculating the perplexity value of the number of topics of two to thirty topics. In checking the number of topics to be used, the authors calculate the perplexity with a different alpha value with a range of 0 to 1. From the smallest value generated for each alpha, the authors look for the smallest value overall. Then it is found that the smallest perplexity value is owned by the calculation result that applies alpha of 0.2, which is 997.5468. This perplexity value is from 29 number of topics. From the experiments conducted, the greater the alpha, the greater the resulting perplexity value. The details of the smallest value for each alpha can be seen in Table 1.

TABLE 1. Minimum perplexity value for each alpha

| Alpha | Minimum perplexity value |
|---|---|
| 0.2 | 997.5468 |
| 0.5 | 1018.751 |
| 0.8 | 1063.055 |
| 1 | 1067.316 |

With that number of a topic and alpha that found, we can do the Gibbs sampling algorithm to do the topic modelling method. The data that has been cleaned is called into the program and then converted into characters to make data processing easier. Then check each line with the condition whether the line contains text with a character length of more than zero or the line is not empty. Then the results of the checking process are stored in a variable. Then data processing is carried out in the form of eliminating repeated data. First, the data that was previously loaded is converted into a dataframe form. Then each character contained in the text is tokenized and stored into a variable. Then the data processing process is carried out in the form of removing repeated words that are accommodated in variable docs. Then the data processing is carried out in the form of replacing each word into the form of ID. This is done to facilitate the calculations that will be carried out in the next step. The steps taken will set the parameters that will be used for the calculation, namely the number of topics (K), alpha, beta, and iterations. This parameter is adopted from Jacobi et al. [8]. The function of LDA calculation is developed from [21]. The next step in this research is to conduct experiments using the topic modelling method using the Gibbs sampling algorithm. The first step taken is to call data that has been pre-processed before.

The first step in the Gibbs sampling calculation is to initialize the calculation matrix. The matrix to be used is the word-to-topic matrix, the topic assignment matrix, the matrix used to store the number of words assigned to each topic for each document. Then perform repetition to assign topics to each word randomly. Then it is repeated to count the number of words in a document assigned to each topic. After the process is complete, the Gibbs sampling algorithm is calculated according to the formula in Section 3, in Formula (1). This calculation is repeated as many as iterations are performed. At the end of each iteration, an update of the word assignment to a topic is performed and increments the word assignment matrix to the topic and the matrix used to store the number of words assigned to each topic for each document with the results of the new calculations performed. Then the word distribution probability calculation for each topic is carried out as in Formula (2). Then the topic probability calculation is carried out for each document as in Formula (3).

After the calculation has been carried out as many iterations as there are, the next step is to search for the topics assigned to each document. This process is carried out using the principle that the topic that is defined in each document is the one with the greatest probability value. Then a search for word distribution on each topic was conducted. The process carried out is to create a function that contains a matrix that is used to accommodate each word that will be obtained for each topic. To obtain related words on each topic, a search for words with the highest probability value was carried out. Then the function call is carried out by determining the number of words desired. Then, the final step is to keep a list of news and words related to a topic. To save the result of this research, a function is made. In that function, there is a loop from one to as many topics as specified. Then form a list that contains a list of news and words related to the topic.

Then determine the place where the file will be stored and determine the name of the file. Then the function is called and the file will be created in the specified directory.

There are twenty-nine topics with fifteen related words in each topic. Based on those words, we can conclude the name of the topic as we can see in Table 2.

4. **Discussion.** In this research, we used some of the pre-processing syntaxes in library R, there are tm and topicmodels. We also use some code to remove noise in the data and we use stopwords collection from the database in *https://www.ranks.nl/stopwords* [20]. With this combination of the method, we can get cleaner data than when we just used the library. After this step, we do a calculation of perplexity value to know the optimal number of topics. Choosing a lower perplexity value can help us to conduct a better list of topics. Determining the number of topics used is as done by Jacobi et al., in their research on the journalistic text. The higher the number of topics is, the lower the resulting perplexity value will be. In their research, there was a significant decrease up to topic twenty-five. At around topics twenty-five to fifty, the decline in the perplexity value was less noticeable. Finally, Jacobi et al. chose twenty-five as the number of the topics that they used [8]. Looking at Jacobi et al.'s research, the authors adopted it in this study. From the smallest value generated for each alpha, the authors look for the smallest value overall. Then it is found that the smallest perplexity value is owned by the calculation result that applies alpha of 0.2, which is 997.5468. This perplexity value is from 29 number of topics. From the experiments conducted, the greater the alpha, the greater the resulting perplexity value. If we look at the results of the author's perplexity calculation, it can be concluded that the correct number of topics is twenty-nine topics.

With this number of a topic that we get, we can do the topic modelling method, resulting list of the topic and the words related to every topic from the data that we analyze. After analyzing the data using the topic modelling method, the results were found as shown in Table 2. From these results, it can be seen that there are still some group of words that cannot be interpreted, such as what happened in Evans' research. In Evans' research, he omitted twelve topics from a total of forty-five topics from the analysis [10]. After filtering the topics, thirty-three topics were labeled according to their related words. In the results that appear in this study, there are also several topics that contain words that are less related. So it is difficult to label the name of the topic. In this study, the authors concluded the resulting topics by distributing questionnaires to people. This is done to minimize the level of subjectivity that might occur. Decisions are taken after the questionnaire has been filled in and the percentage is calculated. Based on the results of filling out the existing questionnaires, there were 8 topics that were formed based on the analysis carried out. The topics are crime, economy, health, coronavirus, lifestyle, humanity, US election, and politic. The most appearing topic is coronavirus as we can see in topics 4, 5, 17, 20, 21, 23, 24, 26, and 29. This happen because the data is from 2020, where the virus is spread widely in every country. This virus also affects many fields of human life. So, most of the news is about coronavirus.

This paper uses a different approach to grouping news based on the themes and topics discussed. By automating the process of grouping news according to the appropriate topic, this can increase the efficiency of the time it takes compared to distributing news to each field. Although the results in this study still need development in order to truly classify news based on the right topic, this research can be a starting point for automating news grouping by applying the concept of topic modeling. This research focused on shortening the time used to divide a lot of news into its relevant field compared to doing this task manually. This new method of grouping news will be of great use to the redactors and the reviewers of news in their organization. People who work in a television station, especially

TABLE 2. Result

| Topic | Related words | Label |
|---|---|---|
| Topic 1 | Return, tell, row, tax, black, royal, release, teacher, honour, killer, rare, boy, accuse, wrong, poison | Crime |
| Topic 2 | Ban, London, bank, travel, finance, file, list, hotspot, visit, music, save, account, Italy, trust, suspend | Economy |
| Topic 3 | Coronavirus, UK, case, rise, week, record, consider, high, infection, hospitality, extend, rate, Spain, surge, month | Health |
| Topic 4 | Coronavirus, school, government, pandemic, amid, health, quarantine, fear, meal, pupil, free, Australia, spike, red, Europe | Coronavirus |
| Topic 5 | COVID, restriction, England, Johnson, boris, Starmer, tight, measure, call, half, area, northern, question, defend, announce | Coronavirus |
| Topic 6 | COVID, pub, close, time, shut, UK, update, Christmas, restaurant, curfew, closure, drive, mayor, open, September | Lifestyle |
| Topic 7 | Test, work, app, virus, trace, John, fire, global, fund, clear, sex, staff, share, baby, crisis | Humanity |
| Topic 8 | Call, protest, woman, change, hold, death, action, jail, lorry, thousand, essex, rival, critise, victim, review | Humanity |
| Topic 9 | COVID, hospital, will, vaccine, day, patient, delay, Ferrier, film, parliament, trial, reopen, son, ready | Health |
| Topic 10 | Warn, virus, minister, firm, leave, Sturgeon, scientist, top, help, impact, flood, water, loan, Madrid, buy | Humanity |
| Topic 11 | Die, year, man, hit, murder, age, drug, dead, whale, fight, Armenia, Nagorno, Karabakh, attempt, find | Crime |
| Topic 12 | Attack, break, France, fine, party, star, street, Paris, boss, bid, bad, halt, economic, conservative, British | Economy |
| Topic 13 | Police, officer, shoot, meet, sir, life, save, David, prince, suspect, crime, join, search, car, reject | Crime |
| Topic 14 | Trump, court, house, white, supreme, Ginsburg, prison, China, asylum, reveal, Ruth, Bader, Donald, military, serious | Politic |
| Topic 15 | Manchester, Brexit, face, UK, deal, talk, EU, trade, great, winter, bill, continue, power, mask, shopper | Economy |
| Topic 16 | Death, people, pay, kill, breaker, circuit, crash, live, family, tribute, issue, hope, apologize, young, spark | Humanity |
| Topic 17 | COVID, job, tier, risk, plan, move, support, level, worker, scheme, alert, system, cut, three, set | Coronavirus |
| Topic 18 | Trump, election, test, happen, debate, Biden, claim, vote, presidential, positive, concern, president, labour, clash, promise | US Election |
| Topic 19 | North, inquiry, charge, east, leader, city, mix, help, arena, state, ahead, video, bring, crowd, blast | Lifestyle |
| Topic 20 | COVID, Scotland, student, university, outbreak, late, Scottish, link, exam, isolation, food, sell, cancer, back, confirm | Coronavirus |
| Topic 21 | Lockdown, wale, COVID, rule, locdal, good, service, business, lockdowns, train, rail, national, steal, Poland, country | Coronavirus |
| Topic 22 | Win, child, woman, post, high, arrest, abuse, flu, furlough, online, award, Hankock, prize, launch, probe | Health |
| Topic 23 | COVID, coronavirus, rule, long, care, lose, Liverpool, tough, curb, holiday, wait, increase, worry, hour, term | Coronavirus |
| Topic 24 | Lockdown, COVID, wale, fear, business, drug, rail, national, Swansea, item, cancel, transport, view, halt | Coronavirus |
| Topic 25 | Police, death, officer, shoot, meet, man, suspect, crime, inquiry, tribute, boss, search, dead, centre, son | Crime |
| Topic 26 | Restriction, COVID, coronavirus, England, week, local, tight, lockdown, extend, announce, Cardiff, part, church | Coronavirus |
| Topic 27 | Government, north, firm, health, east, Welsh, Starmer, labour, minister, steal, water, trust, reveal | Politic |
| Topic 28 | Work, people, death, breaker, circuit, city, closure, fund, clear, mayor, sell, hope, cost, mass | Economy |
| Topic 29 | Test, COVID, tell, app, worker, police, positive, trace, promise, urge, isolation, launch, Bridgend, contact, strictly | Coronavirus |

those who work as news sorters and news categorizers also need this new way of doing news grouping.

5. **Conclusion.** With this research, we can know that using a combination method, not just using the library provided can yield a better result. Before we do the topic modelling method, we need to know the optimal number of a topic to form more accurate topics. So, we calculate the perplexity value of each topic and choose a lower value to get a good result. In this paper, we compare perplexity value with different alpha, and it is found that alpha 0.2 have the lowest perplexity value, which is 997.5468. The result of the topic modelling is a group of words in each topic that related to each other. With those words, we can conclude the label of the topic. News that associated in that topics also can be known. This new method of grouping news will be of great use to the redactors and the reviewers of news in their organization. People who work in a television station, especially those who work as news sorters and news categorizers also need this new way of doing news grouping. With news grouping, readers also can easily find news that they want to read. Readers can search by the topics and find related news based on that topic.

**Acknowledgement.** We would like thanks to reviewers which make our paper better. This work is supported by Informatics Department, Universitas Atma Jaya Yogyakarta.

## REFERENCES

[1] M. Tanikawa, What is news? What is the newspaper? The physical, functional, and stylistic transformation of print newspapers, 1988-2013, *Int. J. Commun.*, vol.11, pp.3519-3540, 2017.

[2] N. Newman, R. Fletcher, A. Kalogeropoulos, D. Levy and R. K. Nielsen, *Reuters Institute Digital News Report*, 2017.

[3] A. Usai, M. Pironti, M. Mital and C. A. Mejri, Knowledge discovery out of text data: A systematic review via text mining, *J. Knowl. Manag.*, vol.22, no.7, pp.1471-1488, doi: 10.1108/JKM-11-2017-0517, 2018.

[4] D. Korenčić, S. Ristov and J. Šnajder, Document-based topic coherence measures for news media text, *Expert Syst. Appl.*, vol.114, pp.357-373, doi: 10.1016/j.eswa.2018.07.063, 2018.

[5] C. B. Asmussen and C. Møller, Smart literature review: A practical topic modelling approach to exploratory literature review, *J. Big Data*, vol.6, no.1, doi: 10.1186/s40537-019-0255-7, 2019.

[6] B. V. Barde and A. M. Bainwad, An overview of topic modeling methods and tools, *Proc. of 2017 Int. Conf. Intell. Comput. Control Syst. (ICICCS 2017)*, pp.745-750, doi: 10.1109/ICCONS.2017.8250563, 2017.

[7] R. Annisa, I. Surjandari and Zulkarnain, Opinion mining on Mandalika hotel reviews using latent Dirichlet allocation, *Procedia Comput. Sci.*, vol.161, pp.739-746, doi: 10.1016/j.procs.2019.11.178, 2019.

[8] C. Jacobi, W. Van Atteveldt and K. Welbers, Quantitative analysis of large amounts of journalistic texts using topic modelling, *Digit. Journal*, vol.4, no.1, pp.89-106, doi: 10.1080/21670811.2015.1093271, 2015.

[9] S. M. C. Loureiro, J. Guerreiro, S. Eloy, D. Langaro and P. Panchapakesan, Understanding the use of virtual reality in marketing: A text mining-based review, *J. Bus. Res.*, vol.100, pp.514-530, doi: 10.1016/j.jbusres.2018.10.055, 2018.

[10] M. S. Evans, A computational approach to qualitative analysis in large textual datasets, *PLoS One*, vol.9, no.2, pp.1-10, doi: 10.1371/journal.pone.0087908, 2014.

[11] K. Bastani, H. Namavari and J. Shaffer, Latent Dirichlet allocation (LDA) for topic modeling of the CFPB consumer complaints, *Expert Syst. Appl.*, vol.127, pp.256-271, doi: 10.1016/j.eswa.2019.03.001, 2019.

[12] H. Xiong, Y. Cheng, W. Zhao and J. Liu, Analyzing scientific research topics in manufacturing field using a topic model, *Comput. Ind. Eng.*, vol.135, pp.333-347, doi: 10.1016/j.cie.2019.06.010, 2019.

[13] J. Ding, S. Azizbek, Y. Sun, P. Tan and F. Wang, Detecting overlapping communities in networks with extremal optimization, *International Journal of Innovative Computing, Information and Control*, vol.17, no.1, pp.355-368, doi: 10.24507/ijicic.17.01.355, 2021.

[14] D. M. Blei, Probabilistic topic models, *Communications of the ACM*, vol.55, no.4, pp.77-84, doi: 10.1145/2133806.2133826, 2012.

[15] T. Porturas and R. A. Taylor, Forty years of emergency medicine research: Uncovering research themes and trends through topic modeling, *Am. J. Emerg. Med.*, doi: 10.1016/j.ajem.2020.08.036, 2020.

[16] J. Guerreiro, P. Rita and D. Trigueiros, A text mining-based review of cause-related marketing literature, *J. Bus. Ethics*, vol.139, no.1, pp.111-128, doi: 10.1007/s10551-015-2622-4, 2016.

[17] D. A. Ostrowski, Using latent Dirichlet allocation for topic modelling in Twitter, *Proc. of the 2015 IEEE 9th Int. Conf. Semant. Comput. (IEEE ICSC 2015)*, pp.493-497, doi: 10.1109/ICOSC.2015.7050858, 2015.

[18] D. M. Blei, A. Y. Ng and M. I. Jordan, Latent Dirichlet allocation, *J. Mach. Learn. Res.*, vol.3, nos.4-5, pp.993-1022, doi: 10.1016/b978-0-12-411519-4.00006-9, 2003.

[19] LuChang-CS, A news crawler for BBC News, Reuters and New York Times, *GitHub.com*, 2020, https://github.com/LuChang-CS/news-crawler, Accessed on Sep. 25, 2020.

[20] *Stopwords*, https://www.ranks.nl/stopwords, Accessed on Oct. 21, 2020.

[21] Ethen8181, machine-learning_LDA_functions, *GitHub.com*, 2016 https://github.com/ethen8181/machine-learning/blob/master/clustering_old/topic_model/LDA_functions.R, Accessed on Sep. 21, 2020.