Utilizing Topic Modelling in Customer Product Review for Classifying Baby Product

Lay Acheadeth
Computer Science Department, Faculty of Computing and Media,
Bina Nusantara University,
Jakarta, Indonesia 11480
lay.acheadeth@binus.ac.id

Misa M. Xirinda
Computer Science Department, Faculty of Computing and Media,
Bina Nusantara University,
Jakarta, Indonesia 11480
misa.xirinda@binus.ac.id

Nunung Nurul Qomariyah
Computer Science Department, Faculty of Computing and Media,
Bina Nusantara University,
Jakarta, Indonesia 11480
nunung.qomariyah@binus.ac.id

***Abstract*—E-commerce is growing at a breakneck pace. As a result, online shopping has increased, which has increased online product reviews. Often, we come across Amazon products with thousands of reviews, and if we look closely we discover that some of them are completely unrelated to the product. In this study, we conducted research on how product review classification can assist in resolving the issue of comments on incorrect items. The method used in this research consists of 4 steps which are, data acquisition, data pre-processing, topic modeling, and text classification. Where Latent Dirichlet Allocation (LDA) was used as our topic modeling technique, and for text classification we used Support Vector Machine (SVM), Logistic Regression, and Multi-Layer Perceptron (MLP) classifiers. We found out that by combining both topic modeling and text classification, a powerful tool for handling this kind of problem was developed. Adding the topic modeling can improve the model's accuracy performance from 0.61 to 0.78. So, we can conclude that the topic modeling was useful in classifying the product reviews.**

I. Introduction

The growth and popularity of social media and e-commerce have brought a large amount of data, which is very valuable for companies that want to better understand the behavior of their consumers. However, there are cases where the user provides a review on the wrong product, resulting in fault analysis, which is a serious issue for large e-commerce companies like Amazon. For example, on the Amazon website, there are a lot of reviews on some products. Only those reviews that pertain the most to the product are shown. However, if you go through all the reviews on a particular product, you may come across reviews that are unrelated to that product. As a result, we are getting fault data and when taken for analysis, it leads to a fault model. However, in the data analysis stage, this kind of data tends to be eliminated, but what if we just do not allow them to be in the first place. This way, it will reduce time and cost in the data cleaning process and also help improve the quality of data which leads to better model performance to implement in the real-world task.

How can we solve the problem? The answer is Natural Language Processing (NLP). Natural Language Processing is a sub-field of linguistics, Computer Science, and Artificial Intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data. Such technologies as Natural Language Processing can divide the text into components so it could understand the context and intent. The machine can then decide which command to execute based on the results of the Natural Language Processing. E-commerce and retail sectors adopted Natural Language Processing among the first. It started with chatbots and conversational interfaces and came to automating business processes and enhancing the consumer experience. In this paper, we are using topic modeling, one of the Natural Language Processing solutions, to address the problem of providing user reviews on the wrong products.

Topic modeling has increased rapidly over the past years as it becomes more important to businesses, especially e-commerce. When a company wants to understand what is being said about it and the reputation of its products online, one of the ways to do this is by using Machine Learning, as well as its sub-area called Deep Learning. Topic modeling is a kind of probabilistic generative model that has been used widely in the field of computer science with a specific focus on text mining and information retrieval in recent years. In technical terms, it is an unsupervised machine learning text mining method, as well as a method of identifying patterns in a cluster. Creating a corpus and running a tool that generates groups of words related to the corpus and distributes them into topics. Topic modeling is also defined as "a method for detecting and tracking clusters of words in large text files".

Since its inception, this model has gotten a lot of attention and piqued the curiosity of academics across a wide range of subjects. Aside from text mining, successful applications have been found in the fields of computer vision, population genetics, and social networks. With topic modeling, businesses can transfer easy tasks onto robots instead of bombarding their employees with data. This way, the employee will have more time to focus on a more productive task rather than every small and insignificant detail which can be achieved by the system.

There are many topic modeling techniques, namely, Latent Dirichlet Allocation (LDA), Non-Negative Matrix Factorization (NMF), Latent Semantic Analysis (LSA), Parallel Latent Dirichlet Allocation (PLDA), and, Pachinko Allocation Model (PAM). However, in this paper, we only focus on one of them which is Latent Dirichlet Allocation [1]. The reason we chose

Latent Dirichlet Allocation (LDA) over the other models is that Latent Dirichlet Allocation (LDA) in comparison to Latent Semantic Analysis (LSI), is more accurate although it takes more time to train. Latent Dirichlet Allocation (LDA) works better with a review or longer text than the Non-negative Matrix Factorization (NMF). Comparing Latent Dirichlet Allocation with Pachinko Allocation Model, Latent Dirichlet Allocation does relate more to the topic extraction modeling and thinking of it independently while Pachinko Allocation Model is more of finding the correlation between topics.

Product classification is being implemented usually either via images or product names. To the best of our knowledge, there is limited research on classifying reviews to the right product category. The goal of product classification is to be able to put products into the right categories. Usually, it is an automated process. For example, scanning the image and saving the product information in a particular category. This product classification is a common process in large malls or big logistic companies. It may sound simple, but it is not. Some products could belong to multiple categories. For example, the item "Men's Basketball Sneakers" could be falling under many categories such as clothing, shoes and accessories, men's shoes, and more. Product categorization is important because it strengthens user experience, improves search relevance, and helps a customer find the correct products. Many algorithms can be used to execute this task, ranging from classic machine learning algorithms such as Naive Bayes, Decision Tree, and Random Forest, to complex deep learning algorithms such as Multi-Class Neural Network and Long Short-Term Memory (LSTM). In this paper, we decided not to implement deep learning and focus more on the classical machine learning approach because our data is limited and the data is structured and predetermined. Another reason is the available hardware and deployment. Machine learning algorithms require less computational power. For example, desktop CPUs are sufficient for training these models. For deep learning models, specialized hardware is typically required due to the higher memory and compute requirements. Specialized hardware is also appropriate because the operations performed within a deep neural network, such as convolutions, lend themselves well to the parallel architecture of the Graphic Processing Unit (GPU).

The product review on its own can be processed by using Natural Language Processing (NLP) pipeline. But, this does not guarantee a good result, because we still capture the general text. Therefore, we decided to combine both topic modeling and product classification to make a more accurate tool, with topic modeling as the input, for solving such problems.

II. Literature Review

There have been multiple different studies conducted by different researchers either on how to extract topics from review, sentiment analysis on the review, or classifying products based on the product name. There is one research related to automatic text classification as well. The first example is research from Dalal and Zaveri [2], on automatic text classification. It is a semi-supervised machine learning task that automatically assigns a given document to a set of predefined categories based on its textual content and extracted features.

The authors in [3] identified various machine learning algorithms that can be used for sentiment analysis on product reviews. These algorithms include Naive Bayes, Support Vector Machine, Decision Tree KNN, Neural Network, and, Random Forest. They concluded that it is necessary to perform the sentiment analysis before launching the new product, as it will help the company know how the customers feel about it before it is released.

Claus and Charles [4] in their research paper about a practical topic modeling technique to exploratory literature review presented an approach not often found in academia. This approach works by using machine learning to analyze literature to identify research paths. The frameworks created had some limitations but the results suggest that topic-modeling-exploratory literature reviews have a promising future.

Bergamaschi, Guerra, and Vincini [5] in their research paper about Data Integration Framework for e-Commerce Product Classification presented an approach suggesting the use of a semi-automatic methodology to define the mapping among different e-commerce product classification standards.

A study on the examination of fake news from a viral perspective was done by Krishnadas, Supriya, Sonia, and, Shahid [6]. And the goal of their research was to investigate the factors that have a major impact on the prediction of fake news. To forecast the likelihood of false news, the researchers looked at a combination of emotion-driven content, emotional resonance, topic modeling, and linguistic aspects of news stories. They found that positive emotions in a text reduce the likelihood of false news. It was also shown that sensational information, such as illicit actions and criminal activity, was linked to false news.

In a paper entitled "Hierarchical multi-label classification using local neural networks", Cerri et al. [7] extended their previous works, where they investigated a new local-based classification method that incrementally trains a multi-layer perceptron for each level of the classification hierarchy. They conducted a thorough experimental analysis, demonstrating that their method achieves comparable results to a robust global method in terms of precision and recall.

With these studies in mind, we decided to conduct our research as we noticed that a lot of them did not involve classifying the review into the right category, usually only the product name classification. Furthermore, to the best of our knowledge, a study that combines topic modeling and classification model is very limited. Last but not least, there is an automatic text classification review which is similar to our work, but we focus on the e-commerce side rather than the automation side which involves more complex tasks to handle. For our research to succeed, we combine topic modeling and multi-label classification to achieve product review classification into categories which is our standpoint. Our goal is to create a model that is capable of detecting

customer reviews and alerting them if they are giving a review on the wrong product. Being able to classify reviews means if the review made by the customer is in the wrong category from the one the model classifies, then the company can alert the user given that the user needs to log in before giving out the reviews. This is one of Amazon's major issues and what they are trying to tackle. At the moment, we still have limited resources in terms of hardware and expertise in that area. Hence, tackling such issues will be considered for future implementation.

III. Methodology

This section gives an overall workflow of the product review classification via topic modeling for Amazon baby products reviews. Figure 1 shows the workflow of our research.

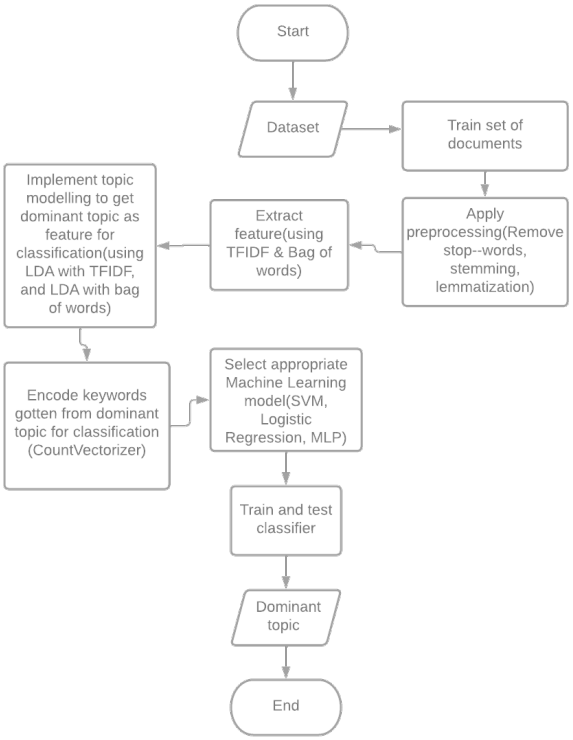

Fig. 1: Research Work Flow

A. Data Acquisition

In this paper, we used a dataset of Amazon baby product reviews which we acquired from Kaggle[1]. The dataset consists of 3 columns and 183,531 consumer reviews on different Amazon baby products. Those 3 columns are name, review, and rating. With this dataset, we explore the possibility of creating a topic modeling model and classify those dominant topics being extracted into the right category after a bit of research. This way, we can make sure all reviews are being classified into the right categories, and if customers give reviews that are not related to the product the model classifies, the company can alert the user given that the user needs to log in before giving out the reviews. The snippet of the dataset is shown in Figure 2 and Figure 3 . The first snippet shows the original dataset, and the second snippet illustrates the used dataset after some cleaning and pre-processing steps.

	name	review	rating
0	Planetwise Flannel Wipes	These flannel wipes are OK, but in my opinion not worth keeping. I also ordered somelmse Vimse Cloth Wipes-Ocean Blue-12 countwhich are larger, had a nicer, softer texture and just seemed higher quality. I use cloth wipes for hands and faces and have been usingThirsties 6 Pack Fab Wipes, Boyfor about 8 months now and need to replace them because they are starting to get rough and have had stink issues for a while that stripping no longer handles.	3
1	Planetwise Wipe Pouch	it came early and was not disappointed. i love planet wise bags and now my wipe holder. it keps my osocozy wipes moist and does not leak. highly recommend it.	5
2	Annas Dream Full Quilt with 2 Shams	Very soft and comfortable and warmer than it looks...fit the full size bed perfectly...would recommend to anyone looking for this type of quilt	5
3	Stop Pacifier Sucking without tears with Thumbuddy To Love's Binky Fairy Puppet and Adorable Book	This is a product well worth the purchase. I have not found anything else like this, and it is a positive, ingenious approach to losing the binky. What I love most about this product is how much ownership my daughter has in getting rid of the binky. She is so proud of herself, and loves her little fairy. I love the artwork, the chart in the back, and the clever approach of this tool.	5

Fig. 2: Original dataset

	review	Category
0	These flannel wipes are OK, but in my opinion not worth keeping. I also ordered somelmse Vimse Cloth Wipes-Ocean Blue-12 countwhich are larger, had a nicer, softer texture and just seemed higher quality. I use cloth wipes for hands and faces and have been usingThirsties 6 Pack Fab Wipes, Boyfor about 8 months now and need to replace them because they are starting to get rough and have had stink issues for a while that stripping no longer handles.	Baby_Care
1	it came early and was not disappointed. i love planet wise bags and now my wipe holder. it keps my osocozy wipes moist and does not leak. highly recommend it.	Diapering
2	Very soft and comfortable and warmer than it looks...fit the full size bed perfectly...would recommend to anyone looking for this type of quilt	Nursery
3	This is a product well worth the purchase. I have not found anything else like this, and it is a positive, ingenious approach to losing the binky. What I love most about this product is how much ownership my daughter has in getting rid of the binky. She is so proud of herself, and loves her little fairy. I love the artwork, the chart in the back, and the clever approach of this tool.	Baby_Care

Fig. 3: Used dataset

B. Data Pre-Processing

1) Data Cleaning: The first step to pre-process the dataset is cleaning it. We did it by dropping the "name" column since we implemented topic modeling, hence we stayed with the "rating" and "review" columns. Afterward, we decided to drop the rating also, and instead of adding one additional column which is the category that was initially blank. Then, we decided to minimize the data usage to 300 rows only. Finally, we annotated those 300 rows of the category's data manually. We could not find any existing dataset for solving this problem. Therefore, we gathered data from various sources and combined them to become a new dataset. We decided to settle with 300 rows only given the time constraint.

Then, we found out that in the first 300 rows we annotated manually, the original data itself is very imbalanced. Hence, we added more data to balance the data more and at last, we have 413 rows, with better-balanced data.

The classes that we used to annotate our data were predetermined. We determined those classes based on the Amazon category website with the following references[2].

[1]https://www.kaggle.com/sameersmahajan/reviews-of-amazon-baby-products

[2]https://www.amazon.com/Best-Sellers-Baby/zgbs/baby-products

The reason why we only used 413 rows of data was first because of the time constraint, this is the data we can come up with since no dataset is available to solve this problem. Secondly, we plan to work with fewer data, to begin with. We want to build a model that works better with small data so it can work even better with big data. In addition, since we are using a machine learning algorithm, it is wise to work with small data first.

2) Tokenization: We split the text into sentences and the sentences into words. And also changed all the letters to lower case and removed punctuation.

3) Stop Words Removal: Then we removed all stop words and words with less than 3 characters.

4) Lemmatization: Words in the third person were changed to the first person and verbs in past and future tenses were changed into the present.

5) Stemming: Words were reduced to their root form.

We then filtered out tokens that appear in less than 15 documents (absolute number) or more than 50% of the documents (fraction of total corpus size, not the absolute number).

6) Feature Extraction: We are using two different feature extraction methods, making a comparison, and using the one with better performance.

First, it is the Bag Of Word (BOW) method. The bag-of-words model is a way of representing text data when modeling text with machine learning algorithms. The bag-of-words model is simple to understand and implement and has seen great success in problems such as language modeling and document classification. In this paper, we implement a bag of words by creating a variable called bow-corpus. Bow-corpus store the data of all the word that occur in certain review. An example has been shown in Figure **??**.

```
Word 13 ("love") appears 1 time.
Word 24 ("book") appears 2 time.
Word 25 ("easi") appears 1 time.
Word 26 ("expect") appears 1 time.
Word 27 ("gift") appears 1 time.
Word 28 ("go") appears 1 time.
Word 29 ("great") appears 1 time.
Word 30 ("help") appears 1 time.
Word 31 ("kid") appears 2 time.
Word 32 ("parent") appears 1 time.
Word 33 ("tri") appears 1 time.
Word 34 ("work") appears 1 time.
```

Fig. 4: Bag of Word (BOW) Result

Secondly, it is Term Frequency-Inverse Document Frequency (TF-IDF). It is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. This is done by multiplying two metrics: how many times a word appears in a document, and the inverse document frequency of the word across a set of documents. It has many uses, most importantly in automated text analysis, and it is very useful for scoring words in machine learning algorithms for Natural Language Processing (NLP). Term frequency-inverse document frequency (TF-IDF) was invented for document search and information retrieval. It works by increasing proportionally to the number of times a word appears in a document but is offset by the number of documents that contain the word. So, words that are common in every document, such as this, what, and if, rank low even though they may appear many times since they do not mean much to that document in particular. In this paper, we implement term frequency-inverse document frequency (TF-IDF) by first converting the bag of word documents into term frequency-inverse document frequency (TF-IDF) in a variable called corpus-tfidf. An example has been shown in Table I.

TABLE I: TF-IDF Result

	Word	**Weight in dataset**
Word 31	kid	0.6101185471094931
Word 24	book	0.3507049773213626
Word 26	expect	0.3170277293126538
Word 32	parent	0.2644307339604643
Word 30	help	0.24761156810927779
Word 33	tri	0.24457566726673635
Word 27	gift	0.23331291836146587
Word 28	go	0.22814285686849048
Word 34	work	0.22814285686849048
Word 25	easi	0.17980086488264918
Word 29	great	0.12142737689587782
Word 13	love	0.08715637917887972

C. Models and Technique

We will be using Latent Dirichlet allocation (LDA) for topic modeling. It is a generative statistical model that allows sets of observations to be explained by an unobserved group that explains why some parts of the data are similar. It is used to classify text in a document to a particular topic. It builds a topic per document model and words per topic model, modeled as Dirichlet distributions [8].

We divided the task into 2 parts. Before being able to classify the review, we first find the dominant topic within the review via topic modeling with Latent Dirichlet Allocation implementation. We used this algorithm with two different inputs, one from the bag of words, and another one from term frequency-inverse document frequency. After getting the *dominant topic*, we try merging the keyword and the dominant keyword as the feature data, with category as the y data. Then, we try dividing the data with 0.1 as the test size and 0.9 as our train size. Finally, we are classifying those reviews based on X to the right categories via three algorithms which are Support Vector Machine, Logistic Regression, and Multi-Layer Perceptron. The hyperparameter settings will be explained more in the result and discussion section along with their performance.

IV. Results and Discussion

After training our models, we used sklearn's built-in accuracy-score method to determine how accurate our classification models are. Aside from the accuracy score from sklearn, we also used a few different metrics, including but not limited to: precision, recall, and F1 scores. Before determining the accuracy score, we need to have the dominant topic first. We

get the dominant topic from the Latent Dirichlet Allocation algorithm that has both words' vectorized values, by using Bag of Words (BOW) and term frequency-inverse document frequency (TF-IDF) calculation, as its inputs.

Overall, the result from Latent Dirichlet Allocation with term frequency-inverse document frequency (TF-IDF) performs better with the three algorithms mentioned, with the best accuracy of 0.79 for Multi-Layer Perceptron. The accuracy of the classifiers is shown in Table II.

TABLE II: Accuracy Scores of the 3 classifiers

	Accuracy Score
Multi-Layer Perceptron	0.79
Logistic Regression	0.76
Support Vector Machine	0.71

Table III shows the results of the dominant topic created by the Latent Dirichlet Allocation model using term frequency-inverse document frequency (tfidf).

TABLE III: F1 Scores of the 3 classifiers

	MLP	**LR**	**SVM**
Activity_entertainment	0.67	0.75	0.75
Apparel accessories	1.00	1.00	1.00
Baby_Care	0.88	0.86	0.77
Baby_stationary	0.71	0.77	0.62
Baby_toddler_toys	0.71	0.71	0.67
Diapering	1.00	0.86	0.86
Gift	0.75	0.57	0.50
Nursery	0.77	0.71	0.77
macro avg	0.81	0.78	0.74
weighted avg	0.78	0.76	0.71

Figure 5 and Figure 6 show the result after the hyperparameter tuning process for Latent Dirichlet Allocation which is the topic modeling, Logistic Regression, Multilayer Perceptron, and Support vector machine for classifying review to the right category.

	Document_No	Dominant_Topic	Topic_Perc_Contrib	Keywords	Text	index
0	0	1.0	0.7760	help, love, camera, play, teether, kid, like, ...	[flannel, wipe, opinion, worth, keep, order, s...	0
1	1	5.0	0.8540	book, monitor, easi, play, pictur, daughter, w...	[come, earli, disappoint, love, planet, wise, ...	1
2	2	0.0	0.8541	look, gift, stick, wall, book, small, daughter...	[soft, comfort, warmer, look, size, perfectli,...	2
3	3	7.0	0.7711	enjoy, play, book, love, littl, learn, year, m...	[product, worth, purchas, like, posit, ingeni,...	3

Fig. 5: if alpha ==’symmetric’ and eta==’auto’

TABLE IV: Support vector machine

	C	**gamma**	**accuracy**
SVM	[0.2,2,20,200,2000]	[2,0.2,0.02,0.002,0.0002]	0.56

The result show two different keywords and a dominant topic due to different feature extraction. However, the Latent Dirichlet Allocation model using term frequency-inverse document frequency (Tfidf) seems to generate a higher accuracy

	Document_No	Dominant_Topic	Topic_Perc_Contrib	Keywords	Text	index
0	0	1.0	0.7167	size, book, kid, play, help, sling, great, lik...	[flannel, wipe, opinion, worth, keep, order, s...	0
1	1	5.0	0.6769	monitor, camera, play, easi, daughter, babi, l...	[come, earli, disappoint, love, planet, wise, ...	1
2	2	0.0	0.5432	look, stick, gift, diaper, buy, wall, small, w...	[soft, comfort, warmer, look, size, perfectli,...	2
3	3	6.0	0.8293	diaper, babi, think, month, year, love, vibrat...	[product, worth, purchas, like, posit, ingeni,...	3

Fig. 6: if alpha ==’0.3’ and eta==’0.3’

TABLE V: Logistic Regression

	param_grid	**n_jobs**	**accuracy**
LR	grid	-1	0.68

TABLE VI: Multi layer perceptron

	hidden_layer_sizes	**max_iter**	**accuracy**
MLP	(250,200,100)	1000	0.69

when generating the dominant topic. In addition, when using both dominant topics from Latent Dirichlet Allocation with a bag of words and Latent Dirichlet Allocation with term frequency-inverse document frequency (Tfidf) for classification, we discover that Latent Dirichlet Allocation with term frequency-inverse document frequency (tfidf) performs way better than Latent Dirichlet Allocation with a bag of words when taking its result to implement multi-label classification.

V. Computational Complexity

The overall result we get is based on different computational complexity. Some have high time complexity but low space complexity, and some have low time complexity and high space complexity. In this paper, we will measure the time complexity only. For Multi-layer perceptron, we receive a **0.79** accuracy within **5.56 seconds** with the test size of **0.1**. For Logistic Regression, we achieve **0.76** accuracy within **117.56 seconds** which is quite long compared to Multi-layer perceptrons. Support Vector Machine achieves **0.71** within **11.86 seconds** which is faster than Logistic Regression in terms of time but lower in accuracy and inferior to Multi-layer perceptrons in terms of time execution and accuracy.

VI. Conclusion and Future Work

We began our research to create an accurate classification model for classifying the review. After seeing the result, we realized that we are not yet close to our goal. As seen from our best performance, the accuracy is only 0.79 which is from the Multi-Layer Perceptrons algorithm with term frequency-inverse document frequency (TF-IDF) as the vectorize technique. That will not be acceptable for working in a real environment. The performance is poor due to the imbalance of our data which we already stated. For our future work, we could use the Neural Network technique on this since Neural Network has more potential in parameter tuning and improving performance. In addition, we can try adjusting Natural Language Processing stemming process. Instead of

getting all types of words, we should only get the noun and verb. That may yield a better result. Finally, we can try to balance the data even more for better model performance.

REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[2] M. K. Dalal and M. A. Zaveri, "Automatic text classification: a technical review," *International Journal of Computer Applications*, vol. 28, no. 2, pp. 37–40, 2011.

[3] H. Paruchuri, S. Vadlamudi, A. Ahmed, W. Eid, and P. K. Donepudi, "Product reviews sentiment analysis using machine learning: A systematic literature review," *Turkish Journal of Physiotherapy and Rehabilitation*, vol. 23, no. 2, pp. 2362–2368, 2021.

[4] C. B. Asmussen and C. Møller, "Smart literature review: a practical topic modelling approach to exploratory literature review," *Journal of Big Data*, vol. 6, no. 1, pp. 1–18, 2019.

[5] S. Bergamaschi, F. Guerra, and M. Vincini, "A data integration framework for e-commerce product classification," in *International Semantic Web Conference*. Springer, 2002, pp. 379–393.

[6] K. Nanath, S. Kaitheri, S. Malik, and S. Mustafa, "Examination of fake news from a viral perspective: an interplay of emotions, resonance, and sentiments," *Journal of Systems and Information Technology*, 2022.

[7] R. Cerri, R. C. Barros, and A. C. De Carvalho, "Hierarchical multi-label classification using local neural networks," *Journal of Computer and System Sciences*, vol. 80, no. 1, pp. 39–56, 2014.

[8] E. Cambria and B. White, "Jumping nlp curves: A review of natural language processing research," *IEEE Computational intelligence magazine*, vol. 9, no. 2, pp. 48–57, 2014.