Classifying Review into Category via Topic Modelling

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

***Abstract*— Ecommerce is growing at a breakneck pace. As a result, online shopping has increased, which has resulted in an increase in online product reviews.**

I. INTRODUCTION

The growth and popularization of social media and e-commerce has 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 user provides review on the wrong product which causes fault analysis and this is a major problem for big E-commerce company such as Amazon.

Hence, is there a way to solve it? One better way is to alert the user whenever they are providing review on the wrong product.

That is possible, but how?

Therefore, topic modelling and product classification is working together to solve such problem. We will first introduce topic modelling.

Topic modelling 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 using Machine Learning, as well as its sub-area called Deep Learning.

So, what exactly is topic modelling? 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 are able to transfer easy tasks onto robots instead of bombarding their employees with data by employing topic modeling. Consider how much time your staff might save and devote to other essential activities each morning if a computer could filter through endless lists of customer surveys, reviews or support problems.

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 two of them which are LDA and LSA.

Now, we will introduce you to product classification. Product classification is being implemented usually either via images or product name. There's barely any research that does on classifying review to the right category yet. Product classification is basically being able to put product into the right categories. Usually, it's an automation process. For example, scanning the image and save the product information in one category. This is very frequent in the large mall or big logistic companies. It may sounds simple, but it's not. There are products that could belong to multiple categories. For example, the item ”Mens Basketball Sneakers” could be falling under many categories such as clothing, shoes and accessories, men shoe's, and more.

Product categorization is important because it's strengthen user experience, improve search relevance, and helps customer find your site.

A lot of algorithm can be used to execute this task, ranging from normal machine learning algorithm such as Naive Bayes, decision tree, random forest, to the complex deep learning algorithm such as multi class neural network and LSVM.

However, although algorithm does not satisfy the real environment since its accuracy is usually so low because of the fact that lots of scenario can happen when categorizing.

Combine everything together, topic modelling and product classification combines will make a great tool for solving such problem.

II. LITERATURE REVIEW

There have been multiple different studies conducted by different researchers either on how to extract topic from review, sentiment analysis on the review, or classifying product based on product name. There's also one research related to automatic text classification as well.

First example is a research from Mita and Mukesh [1] on automatic text classification. It's a semi supervised machine

learning task that automatically assigns a given document to a set of pre-defined categories based on its textual content and extracted features.

The authors in [2] identified various machine learning algorithms that can be used for sentiment analysis on product reviews.These algorithms include Naive Bayes, SVM, Decision Tree KNN, Neural Network, and, Random Forest. They concluded that its 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 [3] in their research paper about a practical topic modelling technique to exploratory literature review presented an approach not often found in academia. This approach works by using machine learning to analyze literature in order 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 [4] 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.

With this 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, there's not yet a study that combine topic modelling and classification model together as well. Last but not least, there's 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 task to handle. For our research to succeed, we combine topic modelling 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 review and alert them if they are giving review on the wrong product. Being able to classify review into category means if the review made by customer is in the wrong category from the one model classify, we alert our user. This is one of Amazon's major issue and what they are trying to tackle.

III. Methodology

This section gives an overall work flow of the review classification via topic modelling for Amazon baby products reviews.

A. Data Acquisition

In this paper we used a dataset of Amazon baby products reviews which we acquired from Kaggle. The dataset consist of 3 columns and 183531 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 modelling model and classify those dominant topic being extracted into the right category after a bit of research.

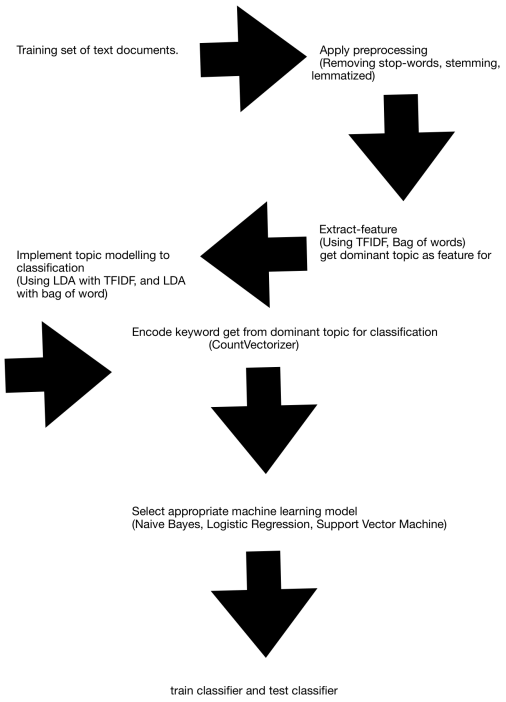

Fig. 1. Work Flow

B. Data Pre-Processing

Has seen in Fig. 1, after we acquire the dataset the following step is data pre-processing. The initial stage in text mining is data pre-processing. Data pre-processing plays an important part in text mining beacause it converts text from human language to machine-readable format. The pre-processing stage is crucial for structuring unstructured material and preserving keywords that may be used to describe text subject categories. Prepositions, pronouns, and other words having no special meaning can be found in natural language text. So, after data acquisition the pre-processing process is divided into some steps.

1) Data Cleaning: The first step to pre-process the dataset is cleaning the data. We did it by dropping the "name" column since we implemented topic modeling, hence we stayed with "rating" and "review" columns. Afterward, we decided to drop the rating also, and instead adding one additional column which is category. It is blank for now. Then, we decide to minimize the data usage to 300 row only. Finally, we annotated those 300 rows of category's data manually.

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

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

4) Lemmatization: Words in third person were changed to first person and verbs in past and future tenses were changed into 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 0.5 documents (fraction of total corpus size, not absolute number).

C. Feature Extraction

1) Bag of words: We decided to implements bag of word because it's the most common method researcher used for testing. It's a simplifying representation in natural language processing and information retrieval. The bag-of-words model is commonly used in methods of document classification where the occurence of each word is used as a feature for training a classifier. In this case, we are trying to figure out how frequent a word is occuring in each review and try to estimate the weight the lemmatized word has with the associated review.

2) TF-IDF: which stands for Term Frequency – Inverse Document Frequency is one of the most essential techniques for representing how relevant a certain word or phrase is to a given text in terms of information retrieval. In this paper we created a "tf-idf" model object using "models.TfidModel" on "bow-corpus" and save it to tfidf, then applied transformation to the entire corpus and called it 'corpus-tfidf'. Finally, we preview TF-IDF scores for our first document.

3) Models and Technique: We divided these into 2 parts. Before being able to classify the review to the category, we first find the dominant topic within the review. For this process, we use LDA model which stands for latent Dirichlet allocation. It's a generative statistical model that allow sets of observations to be explained by unobserved group that explain why some parts of the data are similar. We used this algorithm with two different inputs, one from the bag of words, and another one from tfidf.

After extracting topic from review and get the dominant topic, we implements multi class classification by classifying those dominant topic keyword into categories via three algorithm which are Support Vector Machine, Logistic Regression, and Naive Bayes.

IV. Results and Discussion

After training our models, we used the sklearn's built in accuracy-score method to determine how accurate our classification models are. Before that, we need to have the dominant topic first. We get the dominant topic from LDA algorithm that has both Bag of words and Tf-idf as it inputs. The result show two different keywords and dominant topic due to different feature extraction. However, LDA model using bag of word seems to generate a higher accuracy when generating the dominant topic. In addition, when using both dominant topic from Lda with bag of word and LDA with tfidf for classification, we discover that LDA with bag of word performs way better than LDA with tfidf when taking its result to implement multi label classification. The below result is the result from the dominant topic created by LDA model using bag of word since it performs better.

Fig. 2. topic extraction one row tfidf

Fig. 3. topic overall tfidf

Fig. 4. Topic one bow

Fig. 5. Topic overall bow

The Support Vector Machine classifier returned the accuracy-score of 0.56, where as the Logistic regression classifier returned the accuracy-score of 0.61, and last but not least the Naive Bayes classifier returned the accuracy-score of 0.58.

Aside from accuracy-score from sklearn, we also used a few different metrics, including but not limited to: precision, recall, and F1 scores.

	precision	recall	f1-score	support
Activity_entertainment	1.00	0.25	0.40	4
Apparel_accessories	0.00	0.00	0.00	1
Baby_Care	0.81	0.59	0.68	22
Baby_stationary	0.42	0.42	0.42	12
Baby_toddler_toys	0.54	0.80	0.65	25
Diapering	1.00	0.75	0.86	4
Gift	0.45	0.45	0.45	11
Nursery	0.40	0.36	0.38	11
accuracy			0.57	90
macro avg	0.58	0.45	0.48	90
weighted avg	0.60	0.57	0.56	90

Fig. 6. Support Vector Machine

	precision	recall	f1-score	support
Activity_entertainment	1.00	0.25	0.40	4
Apparel_accessories	0.00	0.00	0.00	1
Baby_Care	0.81	0.59	0.68	22
Baby_stationary	0.50	0.42	0.45	12
Baby_toddler_toys	0.58	0.88	0.70	25
Diapering	1.00	0.50	0.67	4
Gift	0.54	0.64	0.58	11
Nursery	0.50	0.45	0.48	11
accuracy			0.61	90
macro avg	0.62	0.47	0.50	90
weighted avg	0.64	0.61	0.60	90

Fig. 7. Logistic Regression

	precision	recall	f1-score	support
Activity_entertainment	0.00	0.00	0.00	4
Apparel_accessories	0.00	0.00	0.00	1
Baby_Care	0.56	0.91	0.69	22
Baby_stationary	0.67	0.67	0.67	12
Baby_toddler_toys	0.67	0.72	0.69	25
Diapering	0.00	0.00	0.00	4
Gift	0.50	0.36	0.42	11
Nursery	0.60	0.27	0.37	11
accuracy			0.59	90
macro avg	0.37	0.37	0.36	90
weighted avg	0.54	0.59	0.55	90

Fig. 8. Naive Bayes

V. Conclusion and Future Works

We began our research with the goal of creating 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 0.61 which is from logistic regression algorithm. That won't be acceptable for working in a real environment.

We can take this further by trying to add more data to balance the data since our data is too low and imbalance as a matter of fact. We can also try using neural network technique on this since Neural network has more potential in parameter tuning and improving performance.

References

[1] M. Dalal and M. Zaveri, "Automatic Text Classification: A Technical Review", International Journal of Computer Applications, vol.28, No.2, 2011.

[2] H. Paruchuri, S. Vadlamuri, A. Ahmed, W. Eid and P. Donepudi, "Product Revies Sentiment Analysis Using Machine Learning: A Systematic Literature Review", Turkish Journal of Physiotherapy and Rehabilitation, vol.32, No.2.

[3] Asmussen, C. B., Møller, C. (2019). Smart literature review: a practical topic modelling approach to exploratory literature review. Journal of Big Data, 6(1), 1-18

[4] Bergamaschi, S., Guerra, F., Vincini, M. (2002, June). A data integration framework for e-commerce product classification. In International Semantic Web Conference (pp. 379-393). Springer, Berlin, Heidelberg.

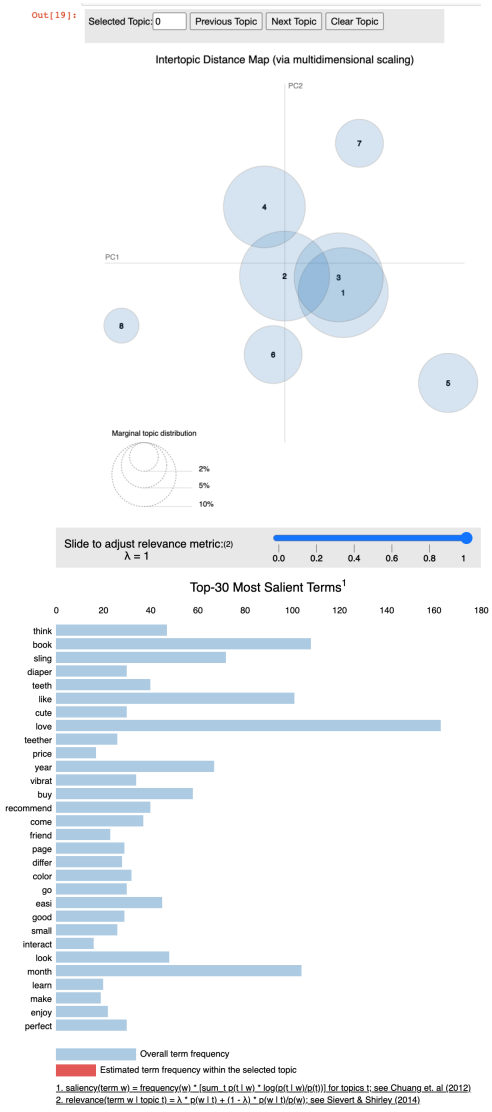

Fig. 9. Intertopic Distance Map