

Finding Customer Behavior Insights for Content Creation in Material and Product Sourcing Using Specialized Topic Analysis

Noptanit Chotisarn¹
Phanuphong Siriphongwatana²
Phattranit Thanisuwiphat²
Sarun Gulyanon³
Winai Nadee^{1,*}

*Corresponding author

¹ Department of Management Information Systems, Thammasat Business School, Thammasat University, Bangkok, Thailand, {noptanit, winai}@tbs.tu.ac.th

² Data Science and Innovation Program, College of Interdisciplinary Studies, Thammasat University, Pathum Thani, Thailand, {phanuphong.sir, laksika.sue}@dome.tu.ac.th

³ Data Science and Innovation Program, College of Interdisciplinary Studies, Thammasat University, Pathum Thani, Thailand, sarung@staff.tu.ac.th

ABSTRACT

In content creation, customer behaviour insights are very important as they help content creators find and create the content that drives sales. To comprehend the customer needs, content creators need not just generalized information but also specific information, which can be different across markets and cultures. This information also needs to have some kind of standard so it can be analyzed in a systematic way. One possible source of this information is the tags based on both customer feedback and the related entities. However, manually analyzing feedback is a time-consuming activity so, in this work, we formulated the topic analysis problem specialized for material and product sourcing. We also compared different text processing and classification methods, which set the benchmarks for reviewing the model performance in the future.

Keywords: Text classification, Customer behaviour insights, Content creation, Specialized topic analysis, Material and product sourcing

INTRODUCTION

Customer behaviour insights are of at most importance in marketing technology since, without this information, the content creators have no clues about the products or services that can attract customers so they can only guess at their best. Without the right content, it is next to impossible to drive sales. The cost of not knowing your customers include spending resources without bringing any value and, in the worst case, losing the customers that we worked so hard to acquire. To comprehend the needs of the customers, content creators need to obtain specific information, e.g., identifying recurrent themes or topics that align with the stakeholders' needs, which can be different across markets and cultures. One way to obtain consumer insights is to analyze the contents that customers usually engage with since they indicate the types of content that customers are interested in.

Since fully manual analysis of contents and articles is a laborious and error-prone process, text classification can be used to organize and understand large collections of text data by assigning tags or categories according to each text's topic, theme, or entities (i.e., products) of interest. Then, the manual analysis is limited to only relevant tags, which makes it become more feasible but still a laborious task as the number of tags can be high, so it is difficult for the analysis to include all the relevant and correct tags. Moreover, the assigned tags must be defined at the right level of specifications; otherwise, if it is too generalized, the tags give no new information, while if they are too specific, only a handful of text data will fall into the categories. We identified this problem as the narrow topic analysis problem.

In this work, we first formulated the narrow topic analysis problem for material and product sourcing in the architecture industry. Our narrow topic analysis involves the classification of text articles based on two types of tags: product categories and themes. This problem is challenging because the product categories tagging is a hierarchical classification, while the notion of themes, in this case, involves concepts that are abstract, vague, and debatable such as styles and trends since they are based on the audience's point of view. As a result, there are no consensus labels for the theme tags.

In order to address the above challenges of the narrow topic classification problem for creating content in material and product sourcing, we adopted the framework as shown in Fig.1, which consists of data processing for standardizing the text, feature extraction for finding the text embedding, and machine learning model for narrow topic classification. Another issue that must

be addressed is the formulation of tags the narrow topic analysis problem. When the task is ill-defined, the machine learning solution can actually be beneficial as it can act as the referee and gives the standard, which everyone can follow, to the problem. However, the key ingredient that is needed for the AI-based solution to work is that all stakeholders (i.e., IT and business units in our case) must settle for the set of tags. All stakeholders do not need to agree on all tags, but each stakeholder must approve that the tags they need are covered.

Hence, in this paper, our contribution includes: (a) a case study of formulating the narrow topic analysis problem in understanding customer behavior for creating content in material and product sourcing, which shows how the problem can be formulated even when the task is vague and ambiguous, and (b) the comparison of methods within the proposed framework for the narrow topic classification.

RELATED WORK

A review on text classification by Minaee et al. (2021) shows multiple applications of text classification tasks including Sentiment Analysis, News Categorization, Topic Analysis, Question Answering (QA), and Natural language inference (NLI). The methods for automatic text classification can be categorized into two groups: rule-based methods and data-driven based methods.

Rule-based methods classify text into different categories using a set of pre-defined rules, which are defined by the experts in the corresponding domains. The main advantage of this kind of methods is the interpretability, which is the ability to explain how the outcome is derived in a comprehensible manner. However, the downfall is that there exist some tasks (i.e., theme tag classification in our case) that even the experts cannot agree on the consensus among themselves so the agreeable rules cannot be derived.

Data-driven based methods or machine learning based methods have gained lots of attention recently. Typical machine learning based models follow the two-step procedure including the feature extraction and classifier. The first step involves computing some hand-crafted features from the article or any other textual unit of interest. The second step is to feed these features to a classifier for making a prediction if they are new/unseen data or feed them to train a classifier if their outputs are known. The popular choices of classification algorithms include Naïve Bayes (John & Langley, 1995) and support vector machines (SVM) (Cortes & Vapnik, 1995).

METHOD

Our method follows the steps shown in Fig.1. First, we defined the tags, which are the expected outputs of the narrow topic analysis. Then, data processing is explained in order to normalize the texts. Next, the feature extraction is discussed to find the representation (i.e., the embedding) of the articles suited for the problem. Finally, the machine learning models are described and compared their results on our task of the narrow topic analysis problem in understanding customer behavior for creating content in material and product sourcing.

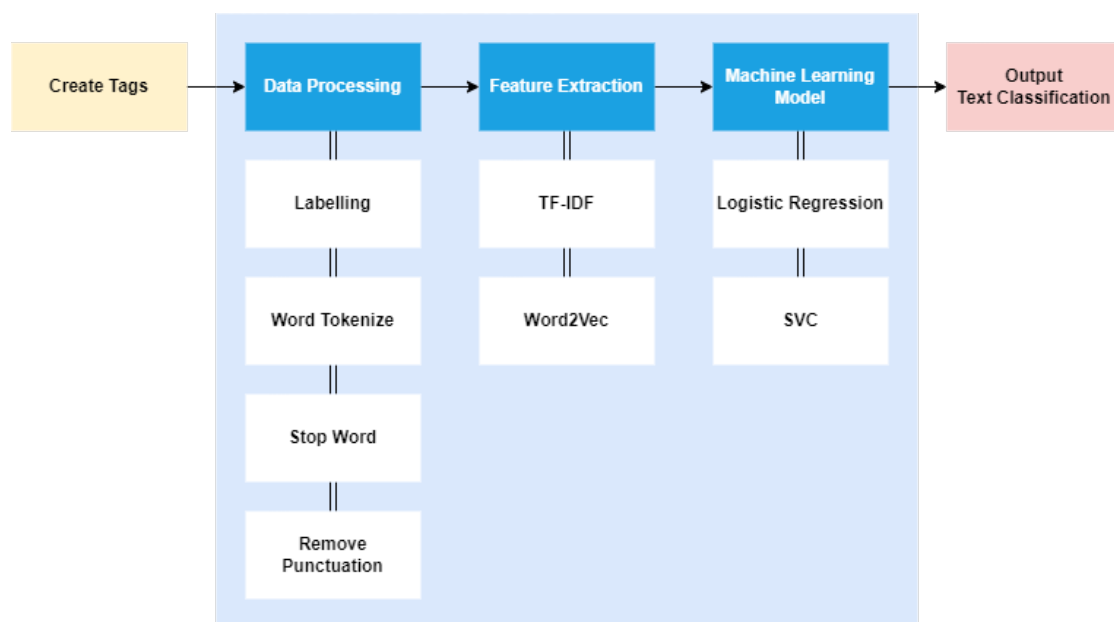


Fig.1 shows the overall steps of our method.

Defining Tags

Two tags sets are needed by the content creator: theme tags and product tags. These two tags are derived from every article. Their definition is given as follows:

- 1) Theme tags are the key words that appear repeatedly in the articles. These tags are then go through the consultation between IT and business units to settle the set of tags that are useful for business unit, while the IT unit keeps track of the complexity of the problem. In our case study of creating content in material and product sourcing, there are 57 tags at the end. The problem is defined as the multilabel classification, where there are possibly multiple numbers of targets for each article, while the target cardinality is two that indicting whether the article is considered in this category.
- 2) Product tags are the tag that classifies construction equipment by material type, shape, and how it used. This information is actually provided by the authors of the articles but as a non-hierarchical tag, which may be incomplete or inconsistent according to the tag hierarchy we adopted. So the classification of product tags are meant to categorize and subcategory articles that are pre-tagged with human resources. The product tag that we adopted in this work consists of 23 categories and 167 sub-categories.

Data Preprocessing

In this section is preparing data to be ready for feature extraction. We perform preprocessing on content and title of the articles. First, load data and fill fields that are not labeled with the number 1, add the number 0 to create a label and detect numbers and links, and label them to eliminate the unimportant parts of the text before training model.

Then start to clean data in 'title' and 'content' columns with th-simple-preprocessor to remove special character. Word tokenization is performed with content column and title column using "newmm" — Dictionary-based Thai Word Segmentation using maximal matching algorithm and Thai Character Cluster (TCC). Word tokenize which are process of splitting a sentence, phrase, an entire text document into smaller units, each of these smaller units are called tokens. Tokens are useful for finding patterns and tokenization helps to substitute sensitive data elements with non-sensitive data elements (Phatthiyaphaibun et al., 2016).

Word Vectorization

It is the process of converting data into a form that can be utilized by the machine learning techniques. Feature extraction aims to reduce the number of features in a dataset by creating new features from the existing ones. In this wok, two feature extraction methods are studied, including TF-IDF and Word2Vec (Mikolov et al., 2013).

- 1) TFIDF or Term Frequency – Inverse Document Frequency is a combination of Term of frequency (TF) and Inverse document frequency (IDF). TF-IDF is a technique that primarily considers the composition of words in document without considering the sequence of words within the document.

Term of frequency (TF) is used to measure how many times a term is present in a document. For example, document "A1" containing 1,000 words and the word "Sky" is present in document "A1" for 25 times. So, the occurrence of term in document is divided by all of word that the document contains. In this example, the term frequency of "Sky" will be $TF = 25/1,000 = 0.025$

Inverse document frequency (IDF) is used to measure the matter of words that appear in all documents. Words appear frequently in all documents tend to be less important. For example, if there are 20 documents and the term "Sky" appears in 5 of those documents. In this example, the inverse document frequency will be $IDF = \log_e(20/5) = 0.602$

- 2) Word2Vec (Mikolov et al., 2013) is a classical method to create word embeddings in the field of Natural Language Processing (NLP). It was developed by Tomas Mikolov and his team at Google in 2013. Word2Vec represents a "word" as a "vector", where these vectors are chosen using the cosine similarity function indicating the semantic similarity between words.

Text Classification Models

In this work, three different machine learning models are studied, including Logistic regression, SVM and Label Powerset with Gaussian Naïve Bayes model.

- 1) Logistic Regression (LR) is selected as the baseline as it is one of the most common and fundamental models for classification problems, with a dependent variable as a discrete variable. Logistic regression uses the sigmoid function to map a predicted values to probabilities.
- 2) SVM (Cortes & Vapnik, 1995) is a supervised machine learning algorithm that can be used for classification or regression problems. They are many possible hyperplanes can be chosen separate two classes of data point. So, the objective of this algorithm is to find the optimal hyperplane in an N-dimensional space, N is the number of features. that distinctly classifies the data points., by N is the number of features. The optimal hyperplane that has the maximum margin, provides some reinforcement so that future data points can be classified with more confidence, in an N-dimensional space that distinctly classifies the data points.

- 3) Label Powerset with Gaussian Naive Bayes (Label-GB) use Gaussian Naive Bayes to classify the hierarchical label seen as a flat label through the powerset. Gaussian Naive Bayes is a variant of Naive Bayes that follows Gaussian normal distribution and supports continuous data. Naive Bayes is an algorithm based on the Bayes theorem. It is a simple classification technique but has high functionality. They find use when the dimensionality of the inputs is high. We use Label Powerset to transform the problem into a multi-class problem with one multi-class classifier, which is trained on all unique label combinations found in the training data.

EXPERIMENTS

The dataset used in the experiments contains 3,139 articles, which is split into training and test data with the ratio of 2:1. In all experiments, for word vectorization, TF-IDF uses all available words after the data preprocessing and Word2Vec uses the pre-trained vector of 300 dimensions publicly available from PyThaiNLP (Phatthiyaphaibun et al., 2016). All possible combinations of two word-vectorization techniques and three classification methods are used to perform theme tag and product tag classification (Label-GB is only applicable to the hierarchical tag like product tags).

The metrics used for evaluation includes precision, recall, and f1-score. Precision (P) is the fraction of relevant instances among the retrieved instances, computed by $P = TP / (TP + FP)$, where TP is true positives and FP is false positives. Recall (R) is the fraction of relevant instances that were retrieved, computed by $R = TP / (TP + FN)$, where FN is false negatives. F1-score is the harmonic mean of precision and recall, which is $F1 = 2PR / (P + R)$.

RESULTS

Table 1 and 2 show the test results of theme tag and product tag classification respectively. The results show that SVM combines with TF-IDF gives the best result for both theme tags and product tags classification. SVM is one of the most popular methods since it usually outperforms other methods and its versatile use in both flat and hierarchical tags. On the other hand, TF-IDF outperforms Word2Vec in this task despite of many works pointed that Word2Vec is better at capturing semantic attributes. Our hypothesis is that Word2Vec needs fine-tuning to our dataset first before it can be effective.

Table 1: Experiment results of theme tags classification.

Techniques	Precision	Recall	F1-score
LR + TF-IDF	0.652	0.2222	0.332
LR + Word2Vec	0.584	0.222	0.280
SVM + TF-IDF	0.658	0.262	0.375
SVM + Word2Vec	0.559	0.180	0.273

Table 2: Experiment results of product tags classification.

Techniques	Precision	Recall	F1-score
LR + TF-IDF	0.9556	0.3443	0.5062
LR + Word2Vec	0.9395	0.2307	0.3704
SVM + TF-IDF	0.9657	0.8692	0.9149
SVM + Word2Vec	0.9269	0.5945	0.7244
Label-GB + TF-IDF	0.8453	0.8397	0.8425
Label-GB + Word2Vec	0.8507	0.8419	0.8463

CONCLUSION

We presented a case study of formulating the narrow topic analysis problem in understanding customer behavior for creating content in material and product sourcing as we started from defining the problem as tags classification and give a guideline for defining the standard tags if none exists. The framework we used for text classification consists of data processing, word vectorization, and classification. We compared two different word vectorization methods and three different classification methods and found that SVM combined with TF-IDF gives the best results on our dataset.

The analysis of tags for finding customer behavior insights are left for future work along with the experiments on the modern classification techniques like deep learning and build on our current project visualization regarding the use of tags as bubble visualization (Chotisarn et al., 2021).

ACKNOWLEDGMENT

This work is supported by the Wazzadu.com for valuable data. The first author wishes to thank Mr. Wissarut Pimanmassuriya for his valuable technical, data support on this project.

REFERENCES

- Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). Deep learning--based text classification: a comprehensive review. *ACM Computing Surveys (CSUR)*, 54(3), 1-40.

- John, G. H., & Langley, P. (1995). Estimating continuous distributions in Bayesian classifiers. In Eleventh Conference on Uncertainty in Artificial Intelligence, Proc., 1995 (pp. 338-345).
- Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3), 273-297.
- Phatthiyaphaibun, W., Chaovavanich, K., Polpanumas, C., Suriyawongkul, A., Lowphansirikul, L., and Chormai, P., "PyThaiNLP: Thai Natural Language Processing in Python," Jun. 2016. [Online]. Available: <http://doi.org/10.5281/zenodo.3519354>
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Chotisarn, N., Lu, J., Ma, L., Xu, J., Meng, L., Lin, B., ... & Chen, W. (2021). Bubble storytelling with automated animation: a Brexit hashtag activism case study. *Journal of visualization*, 24(1), 101-115.