

Classifying Multilingual Text in Quality Assurance

Sze Pei Tan¹, Tien-Ping, Tan²

^{1,2}School of Computer Sciences, Universiti Sains Malaysia, Penang, Malaysia
E-mail: ²tienping@usm.my

Article History: Received: 10 November 2020; Revised: 12 January 2021; Accepted: 27 January 2021;
Published online: 05 April 2021

Abstract : Machine learning systems play an important role in helping and assisting engineers in their daily activities. Many jobs can now be automated, and one of them is in handling and processing customers' complaints before they could proceed with failure investigation. In this paper, we discuss a real-life challenge faced by the manufacturing engineers in a life science multinational company. This paper presents a step by step methodology of multilingual translation and multiclassification of Repair Codes. This solution will allow manufacturing engineers to take advantage of machine learning model to reduce the time taken to manually translate row by row and verify the Repair Codes in the file.

Keywords: text classification, quality assurance, language translation, repair text

1. Introduction

Manufacturing company is one of the agents involved in the creation of sustainable economic growth especially in the middle-income countries (Dan Su, 2016). The major contribution of them is in transferring knowledge from developed countries such as United States to developing countries such as Malaysia, Thailand, Vietnam, etc. However, the question remains if the manufacturing companies in the developing countries are ready, or have adopted industrial revolution of 4.0. This will determine which companies are placed ahead in the competitive industry. The driving technologies to digitalize the manufacturing industry are machine learning, big data, artificial intelligence etc. One of the examples in manufacturing operations is to predict machine failures through sensors and maintenance logs. Defect detection using image recognition will increase the detection rate as high as 90% compared to human inspection – this would prevent any defective products reaching to the customer (Harald Bauer, Peter Breuer, Gérard Richter, Jan Wüllenweber, Knut Aliche, Matthias Breunig, 2017).

In manufacturing, the quality of the product are influenced by the manufacturing techniques (Chrystolouris, 2016), design of the product, incoming materials and human factors (W. Patrick Neumann, 2016) and (Hussain et al., 2017). Therefore, quality assurance department plays a vital role to identify any quality concern and ensure the quality of products. The quality policy in which the company that adopt is to provide the best service and product which fully comply to regulatory requirements. When there is defective product and customer complaints, the failure description data will be available, which eventually leads to failure investigation done by the manufacturing engineers.

However, with global manufacturing sites, the usage of different languages in different countries cannot be avoided. There are more than seven thousand languages spoken as of 2019 (Eberhard, 2019). This introduces the need to use machine translation and machine learning to analyze failure codes in repair text.

2. Problem Description

The data for this study was provided by Agilent Technology, a life science multinational company. The data consists of Annual Failure Rate (AFR) that were logged by the field engineers. An example of the data (not real data) with 6 attributes as below:

Table 1. AFR data

| Product Line | Product Number | Repair Details | Customer Complaints | Repair Class | Repair Office |
|--------------|----------------|-------------------|--------------------------------|--------------|---------------|
| A | AX 123 | Peças substituída | Chemical dispenser not working | RP | US |
| B | AZ 456 | 对齐条形码阅读器 | 由于条形码阅读器未对齐而导致的扫描问题 | AD | China |

Based on Table 1, under the column of Repair Class, “RP” is referring to “Part Replacement” while “AD” is referring to “Part Adjustment”. Often, the Repair Details is entered in different languages and the classification of Repair Class maybe misclassified. Therefore, it is time consuming for the manufacturing engineers to manually translate each row in the file and verified the classification of the Repair Codes.

The new data is generated every month with more than 200 rows of information. It will take hours for manual translation and validation of Repair Codes before the manufacturing engineers can proceed to perform failure investigation. Therefore, there is a need to improve the current routine work of the manufacturing engineers. This study proposed a solution that apply machine learning techniques in manufacturing industry.

3. Materials and Methods

The dataset used consists of 7 years of data with 106 attributes while the total Repair Class (class label) consists of 14 classes. The attributes of the data are not disclosed due to private and confidentiality. The data types include numerical, categorical, nominal, text, date and null values. Figure 1 below shows the steps:

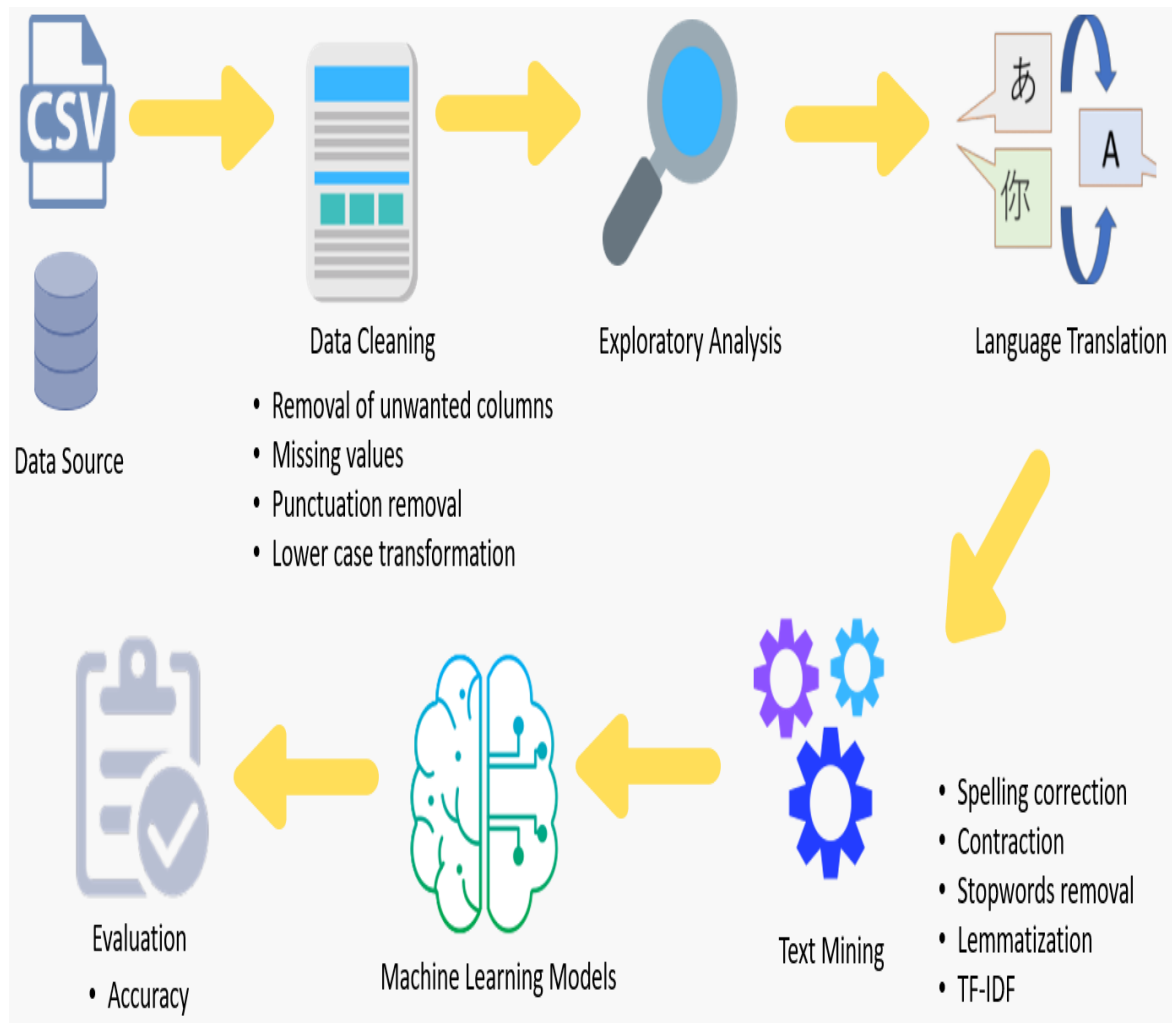


Figure 1.Methodology

Based on Figure 1, after the raw data is extracted from the system, data cleaning is done to remove unimportant columns and missing values. An exploratory analysis is performed to visualize the data to get a better understanding of the data and to gain insights. For example, one issue of main concern to the stakeholders is the product and the manufacturing plant that have the highest repairs because the number of repairs will affect the quality of the product and cost.

After the raw data is standardized and translated to English, the text processing step can proceed with removal of punctuation, spelling correction and normalization. This also includes tokenization, removing duplicate sentences and special character after the data is translated using Google Translate (Tomas Mikolov, 2013).

With the translated text, the text mining technique is performed to create a Word Cloud. This will provide a quick glimpse of what are the frequent words or key points in the Repair Details.

After that, a classifier is created using machine learning algorithms such as Support Vector Machine,

Decision Tree, Naïve Bayes, kNN, Vsrest Classifier, Bagging, Logistic Regression and Rocchio Classifier. In this study, we evaluate the classification results using accuracy, f1 score, precision and recall. The proposed methodology will save manufacturing engineers’ time in analyzing repair codes to allow them to focus on a more complex problem in production.

4. Results and Discussion

The first objective is to automate language translation to English for “Repair Details” column as Repair Codes (class) will be assigned based on the repair details. Only 7 attributes of the data were used for our analysis. The data was visualized using matplotlib libraries. The language translation was performed using Urllib package. Urllib package was used to handle URL request and send – GoogleTranslate website to translate each rows of the file (Python Software Foundation, 2020).

Machine translation using human evaluations are expensive and time consuming – months to complete depending on the size of the data. The data that was translated consists of 4347 rows. The data consists of 23 languages such as English, Spanish, Japanese, Korean, Mandarin, French, Portuguese, Swedish and others. Therefore, a random of 5% of data that consist of short and long sentence were chosen to be manually translated by manufacturing engineers. The human translated text was used as reference to compare with the output of machine translation results.

BLEU was used to evaluate the text translation quality. The BLEU has a value between 0 and 1. When a value of 1 is attained in conjunction to the reference text, it means the translated output is identical to the reference (Kishore Papineni, 2002). The translated text and reference were tokenized before calculating the BLEU score. By default, the BLEU was calculated by the cumulative of 4-gram. The BLEU obtained was 0.114. This lower BLEU value obtained was expected because the manufacturing engineers who translated the text have added more additional information and details to help the engineers to classify the Repair Code manually. After translating the text into English languages, text mining techniques are applied – Natural Language Toolkit (NLTK) and TextBlob packages. Below table is the text mining techniques used:

Table 2. Text Mining Techniques and Functions

| Techniques | Functions |
|---------------------------|---|
| Contraction | Expanding English contraction to reduce the dimension of the text before generating vectors. Contractions pose as a problem for text analytics because there is an apostrophe character in the word. It may become two new words when tokenize (Sarkar, 2016) |
| Lemmatization | Conversion from word to its root word – do not strip suffices but uses vocabulary and perform morphological analysis to get a root words. Lemmatization is preferred over stemming |
| Stop words removal | Remove insignificant words such as “the”, “a”, “me” and others. The text will be checked to ensure actual context of the sentence given is not lost. |
| TF-IDF | Determine importance of a word based on the statistic method. The higher the number of times appear on the document, the higher the value will be |
| Word Cloud | Visualize frequent words appear in document – verbs and pronunciation provide a strong representation of key words in text (Yanqiu Chen, 2019) |

The second objective of the study is to classify the records in the AFR data. The attributes used were the translated Repair Details, Product Number, RESOLUTION_CODE and Repair Class (label). The translated Repair Details, Product Number and RESOLUTION_CODE have been combined into 1 column as the attribute.

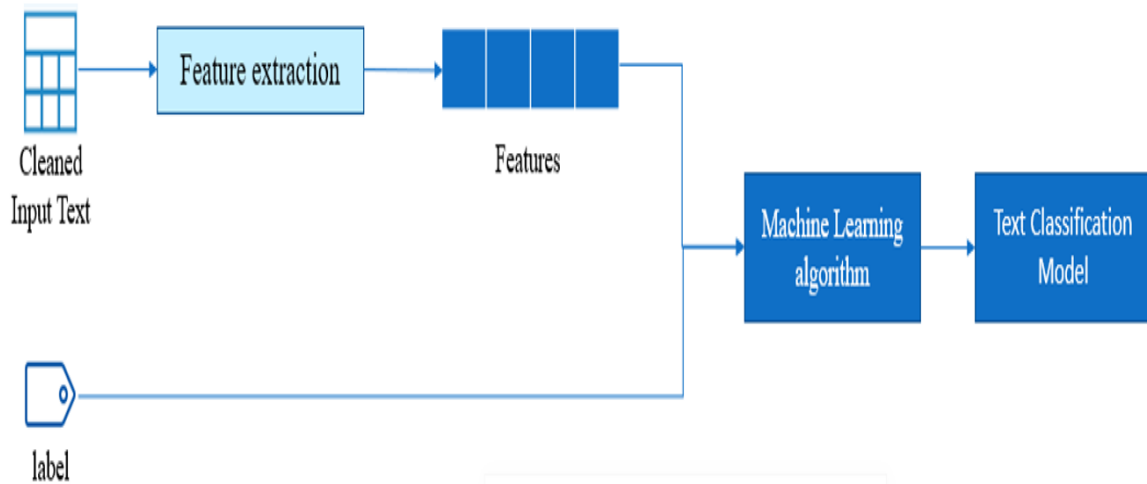


Figure 2. Text classification process

Figure 2 above depicted the text classification step. The training and testing were in 70:30 ratio. Feature extraction using TF-IDF technique was performed to transform each text into a feature set in the form of bag of words vector because texts cannot be interpreted by classifiers directly (Sebastian, 2002). Besides that, the feature selection can be used to reduce the dimension of the features (Yanqiu Chen, 2019).

Table 3. Vocabulary

| | |
|------------------|---------------------------|
| (0, 4706) | 0.2733002302914989 |
| (0, 4097) | 0.24547395010274908 |
| (0, 3696) | 0.5144904247994133 |
| (0, 3691) | 0.2772151524419183 |
| (0, 3440) | 0.2785677354901321 |
| (0, 2070) | 0.14147762058940846 |
| (0, 1823) | 0.4291702880125727 |
| (0, 968) | 0.49162775500713385 |
| (1, 4460) | 0.3234440721629034 |
| (1, 3959) | 0.25507949120071155 |
| (1, 3808) | 0.28077406657564585 |

Table 4. Vectorized text

| Vocabulary | TF-IDF |
|-------------------|---------------|
| replaced | 3600 |
| pump | 3375 |
| mount | 2850 |
| tested | 4460 |
| installed | 2407 |
| board | 1061 |
| setting | 3966 |
| software | 4097 |
| equipment | 1423 |
| power | 3282 |

Table 3 shows snippet of the vocabulary learned from the corpus, and Table 4 shows the vectorized data. The performance of the classifiers was compared by using accuracy, F1 score, precision and recall (Sebastian, 2002). See Figure 3. The top classifiers were Decision Tree, Random Forest, SVM, and Rocchio Classifier. The best classifier is the Decision Tree with accuracy, precision and recall of 0.98 and 0.94 and 0.97. However, there is a possibility of a fully-grown tree that is prone to overfitting because the branches may be specific to the trained data (Sebastian, 2002). To achieve a balance of both high accuracy, precision and recall, Rocchio Classifier is selected and preferred – this is supported through the cheap computation power compared to other classifiers (Kamran

Kowsari, 2019).

Rocchio method is based on the nearest centroid that calculates the similarity between test documents and each of the prototype vector through averaging the vector of the training documents. Rocchio classifier obtained 0.96 in term of accuracy compared to the other top three algorithms such as Decision Tree, Random Forest and SVM. The lower accuracy of Rocchio Classifier in this study was supported by other literatures that the accuracy of the Rocchio classifier was indeed lower than other algorithms that are more computationally complex classier (Sebastian, 2002),(Moschitti, 2003), (Yang, 1999).

The bagging and kNN classifiers have almost similar readings for accuracy, precision, recall, and F1-Score. kNN does not divide the document space linearly unlike the Rocchio classifier. Bagging was used by combining several Decision Trees to reduce bias while increasing the prediction. However, Bagging performed poorer compared to Decision Tree and Random Forest.

Naïve Bayes performed poorer in term of accuracy, precision and recall compared to most of the other algorithms. The classifier was hailed as “the punching bag of classifiers” because it was at the last position in many classification papers (Lewis, 1998)(Tong Zhang, 2004). However, Naïve Bayes is still being used because it is easy and straightforward to implement. The low accuracy of Naïve Bayes is due to systemic issues as such one specific class has more training examples than the other, thus poor weights selection is done for the decision boundary (Jason D. M. Rennie, 2003).

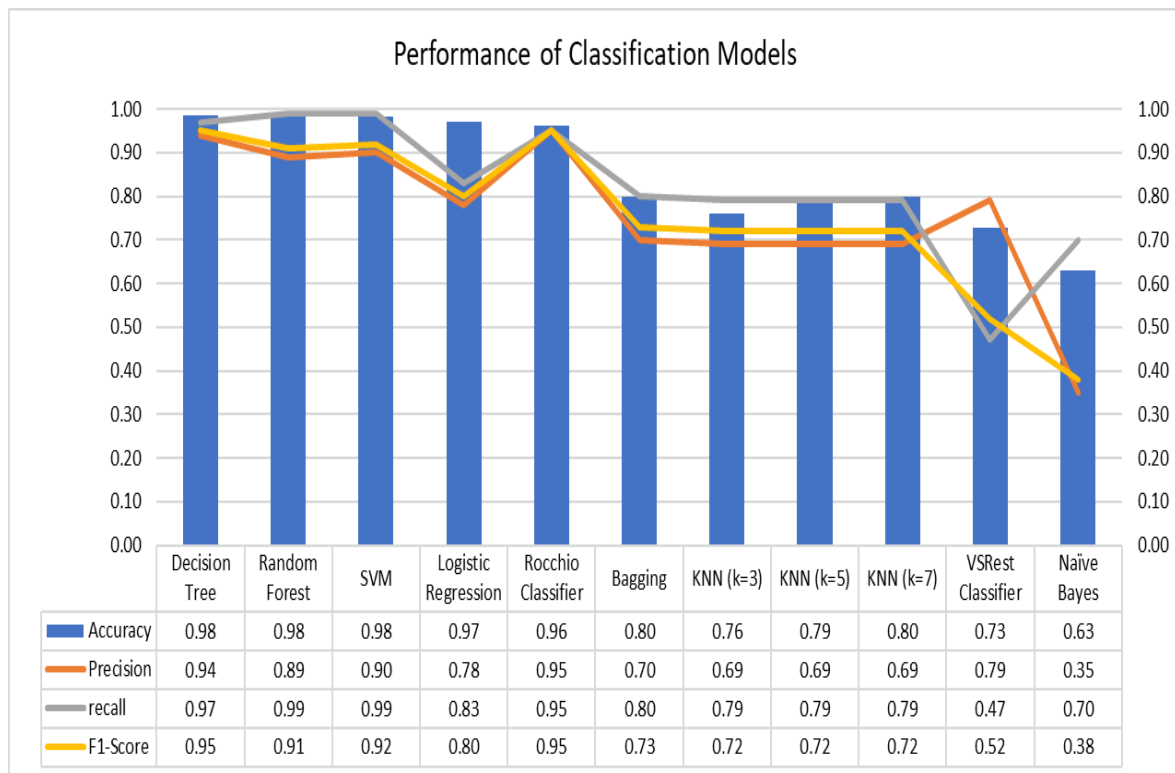


Figure 3. Performance of classification models

5. Conclusion

With the emergence of big data, immense research has been performed in the field of text analytics especially in language translation and text classification. The paper applies machine translation using Urllib package that uses Google Translateto translate the multilingual text to English. The lexical and semantic errorsfrom the translation maypresent, nevertheless, our experiment show that very good classifier can be created using the translated text.

Theclassifierswere built with cleaned and multilingual translated dataset. All attributes were transformed into numerical values before being used to train the models. The classifier selected is RocchioClassifier based on the result inaccuracy, precision, and recall. Manufacturing and quality engineers may no longer need to translate each row manually to understand and classify repair codes manually. The model is able to classify the repair

codes in a limited time. Moreover, this project can be set as blueprint and be proliferated to other teams

6. Acknowledgment

The authors would like to thank Agilent Technologies, Penang, Malaysia for providing the support and data for this project. The authors also thank Universiti Sains Malaysia for the financial support [Project Number = 304.PKOMP.6316283].

References

1. Chrystolouris, H. B. S. 2016. Additive manufacturing methods and modelling approaches: a critical review. *The International Journal of Advanced Manufacturing Technology*: 389-405.
2. Dan Su, Y. Y. 2016. Manufacturing as the key engine of economic growth for middle income economies. Tokyo, ADBI Working Paper Series.
3. Eberhard, D. M., G. F. Simons and C. D. Fennig (eds.). 2020. *Ethnologue: languages of the world*. Twenty-third edition. Dallas, Texas: SIL International.
4. Rennie, J. D. M., L. Shih, J. Teevan and D. R. Karger. 2003. Tackling the poor assumptions of naive Bayes text classifiers. *Proceedings of ICML-2003*. Washington: 616-623.
5. Kowsari, K., K.J. Meimandi, M. Heidarysafa, S. Mendu, L.E. Barnes and D.E. Brown. 2019. Text classification algorithms: a survey. *Machine Learning on Scientific Data and Information*, 10(4): 1-68.
6. Papineni, K., S. Roukos, T. Ward and W.-J. Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. Philadelphia: 311-318.
7. Lewis, D.D., 1998. Naive (Bayes) at forty: the independence assumption in information retrieval. Berlin, Heidelberg, Springer: 4-15.
8. Moschitti, A. 2003. A Study on optimal parameter tuning for rocchio text classifier. Berlin, Heidelberg, Springer: 420-435.
9. Hussain, A., Mkpjojogu, E.O.C., Jamaludin, N.H., Moh, S.T.L. (2017). A usability evaluation of Lazada mobile application. *AIP Conference Proceedings*, 1891, art. no. 020059.
10. Keikha, M., N.S. Razavian, F. Oroumchian and H. S. Razi. 2008. Document Representation and Quality of Text: An Analysis. In: Berry M.W., Castellanos M. (eds) *Survey of Text Mining II*. London, Springer.
11. Nelli, F., 2015. *Python Data Analytics*. New York. Apress.
12. Rosaline, R. A. A. and R. Parvarthi, 2015. Performance analysis of various text algorithms. *International Journal of Pure and Applied Mathematics*: 625-634.
13. Sarkar, D., 2016. *Text analytics with python*. Bangalore, Karnataka, Apress.
14. Sebastiani, F., 2002. Machine learning in automated text categorization. *ACM Computing Surveys*: 1-47.
15. Mikolov, T., Q.V. Le and I. Sutskever. 2013. Exploiting similarities among languages for machine translation.
16. Zhang, T. and F. J. Oles. 2004. Text categorization based on regularized linear classification methods. *Information Retrieval*, 4(1): 5-31.
17. Neumann, W.P., A. Kolus, Richard W. Wells. 2016. Human factors in production system design and quality performance – A Systematic Review. *IFAC-PapersOnLine* 49(12): 1721-1724.
18. Yang, Y., 1999. An evaluation of statistical approaches to text categorization. *Information Retrieval*, 1(1-2): 69-90.
19. Chen, Y. and P. Sun. 2019. A Chinese text classifier based on strong class feature selection and Bayesian algorithm. *Proceedings of ICITBS, Changsha*: 540-543.