Sentiment Analysis Based on Machine Learning Algorithms: A Comprehensive Study

Song Jiang*[1], Ela Kumar[2]

1 Department of Biochemistry, Huzhou Institute Of Biological Products Co., Ltd. China

2 Department of Computer Science and Engineering, KLEF, Vaddeswaram, India

songjiang@hzbio.net

Abstract: The Yelp Dataset comprises data collected from 8,021,122 reviews and 209,393 businesses located in 10 major metropolitan areas. This comprehensive dataset includes multiple aspects related to the businesses. We are interested in assessing the reliability of Yelp's review sentiment algorithm by constructing our own specific sentiment analysis algorithm using data mining and machine learning techniques. The system, based on Natural Language Processing (NLP), generates structured text, followed by the application of machine learning (ML) techniques to classify the text as either a 'good' or 'bad' indicator, used for sentiment prediction. The ML models we utilized here include logistic regression, random forest, k-nearest neighbors, and naive Bayes. Our results demonstrate that three of these models can precisely classify the text and accurately predict sentiment.

Keywords: Sentiment, Machine Learning, NLP, Classification

Introduction

Sentiment analysis, also known as opinion mining, is a natural language processing (NLP) technique used to determine the sentiment or emotional tone behind a piece of text.[1] It has been applied to various areas, including health diagnosis, environmental chemical safety, academia, and more.[2-9] To find an accurate model for sentiment analysis, here, we utilized the data from Yelp to train various machine learning algorithms and evaluate their effectiveness in predicting sentiment. Yelp, a leading company in local business recommendations in North America, reported significant user engagement. As of the second quarter of 2019, Yelp disclosed in their 2019 Q2 Shareholder Letter that they had an average of 61.8 million unique visitors via desktop computers and 76.7 million via their mobile website each month. According to their Investor Relations page as of June 30, 2019, Yelp hosted 192 million reviews on its site.[10] Leveraging this massive user base, business information, and reviews, Yelp has amassed a substantial dataset and developed its own business recommendation system. However, given the subjective nature of recommendations, ongoing efforts persist in exploring methods to enhance and create a more comprehensive recommendation system.[11, 12]

The dataset utilized in this project is readily available from the Yelp Challenge at Kaggle.[13] This dataset includes comprehensive information about businesses, users, and their reviews across ten metropolitan areas spanning four countries. Consisting of a total of six datasets, our primary focus for model construction will be on checkin.json, business.json, and user.json. The largest dataset, review.json, contains over 4.7 million reviews posted on Yelp from 2004 to 2017. With a size of 3.82 GB, it serves as an ideal dataset for practicing NLP. Each review within this dataset is linked to a business ID, user ID, date, star rating (on a scale of 1 to 5), review ID, and its original text. All IDs within this dataset are represented by randomly assigned combinations of numbers and letters to safeguard user privacy while functioning as unique identifiers.

Methodology

The dataset utilized in this article is from Kaggle. The primary approach of our system, as seen in Figure 1, is to take data sets as input to preprocess, followed by NLP processing and analyzing before applying machine learning classification algorithms (logistic, random forest and k-nearest neighbors).[14-17]

For each algorithm, NLP technique is applied, such as stemming, computing occurrence of words, etc. After that, each model is built and implemented, optimized and compared.

Three models are compared according to their performances. Equations 1 and 2 present how these values are obtained based on the predicted class types. Precision (equation 1), recall (equation 2) are used to evaluate the classification performance.

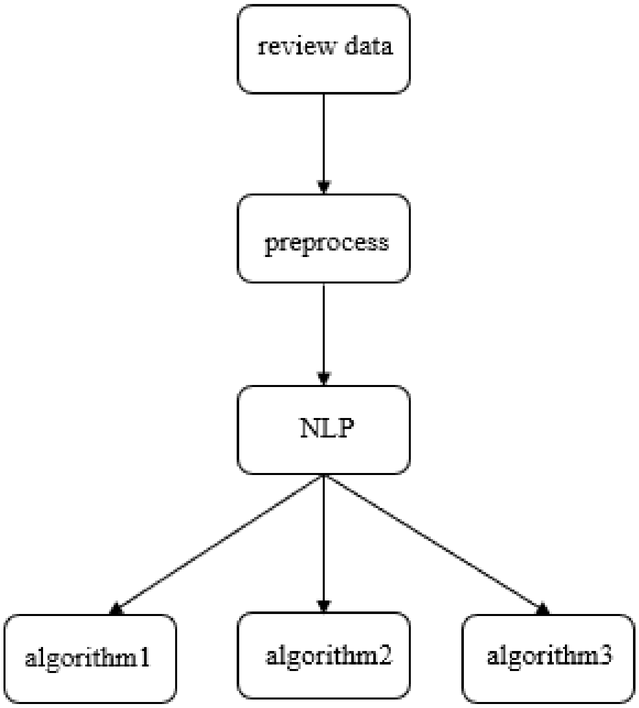


Figure 6. System Overview

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive	False Negative
	Negative	False Positive	True Negative

Table 1: Outcomes for Binary Classification

$$precision = T\,P/(T\,P + F\,P) \tag{1}$$

$$recall = T\,P/(T\,P + F\,N) \tag{2}$$

A. NLP

Before applying algorithms, the text reviews are first processed with word2vec, and then transformed with term frequency–inverse document frequency with Tfidf Vectorizer.

B. Classification

After considering many different classification algorithms, logistic regression, random forest and k-nearest neighbors were chosen, due to their excellent binary classification properties.

Experiments and Results

A. Preprocessing

After data exploration analysis, the review data first transformed from json to csv, followed by reorganization (‘funny’, ‘useful’ and ‘cool’). After that unnecessary columns (‘unnamed: 0’, ‘business_id’, ‘date’, ‘review_id’, ‘type’, ‘user_id’) are removed. After that, each review was calculated according to its sentiment using TextBlob, followed by calculating review stars.

B. NLP Processing

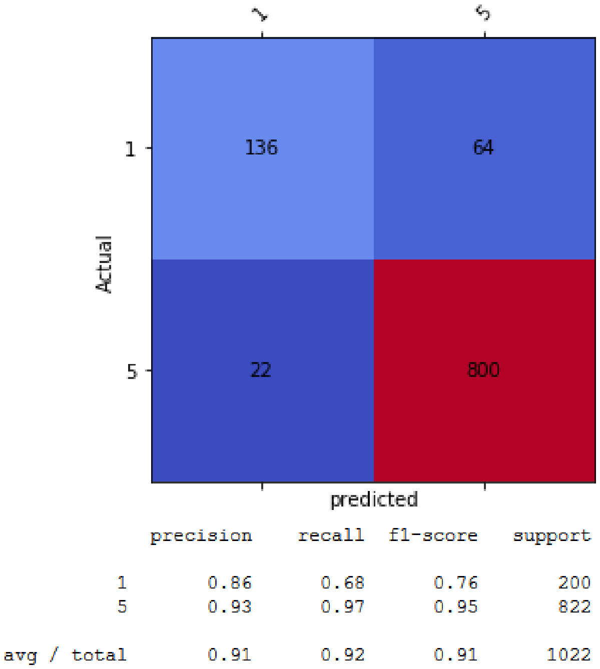
Sentiment prediction models, including Logistic Regression, Random Forest, K-Nearest Neighbors, and Naive Bayes classifier, are implemented to predict Yelp review sentiment. The parameters for each model are optimized to achieve good prediction performance. Specifically, the following steps are followed: 1) Split the data into training data (for model selection) and test data (for model validation), 2) Use natural language processing techniques (apply stemming to text, compute word occurrences or term frequency–inverse document frequency to transform text notes), 3) Implement and optimize popular text classifiers: Logistic Regression, Random Forest, K-Nearest Neighbors classifiers, 4) Compare the performance of the optimized classifiers trained with word occurrences or term frequency–inverse document frequency.

To test a model, the data is initially separated into training (75%) and testing (25%) datasets. The training data is utilized for both model training and cross-validation purposes. The testing data is employed to assess the final performance of each model. As the data comprises text, it needs to be transformed before applying to a model. For this purpose, the historical notes are tokenized with stemming and converted into word occurrences using the sklearn CountVectorizer, or into term frequency–inverse document frequency using the Tfidf Vectorizer.

Parameters such as n_estimators, criterion, and random_state for the random forest were tuned using

GridSearchCV. The best parameters for count vectorizer and tfidf vectorizer were provided, respectively. The steps for tuning parameters for the Logistic Regression and K-Nearest Neighbors models are also the same as those for the Random Forest model

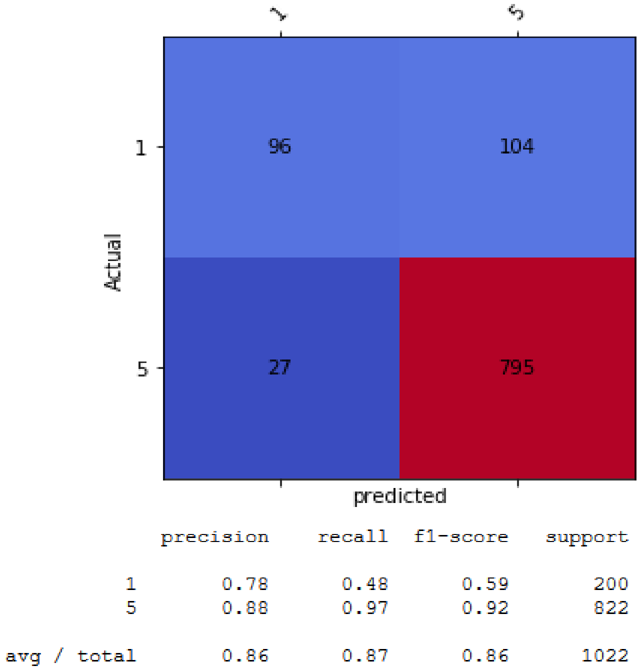
The prediction of star ratings using Logistic Regression, Random Forest, K-Nearest Neighbors, and Naive Bayes is displayed in figures 7, 8, 9, and 10. The best models are Logistic Regression and Naive Bayes, as they have the same precision, recall, and f1-score. Additionally, they both demonstrate the same level of accuracy. Conversely, the K-Nearest Neighbors model exhibits the worst performance.



```
             precision    recall  f1-score   support

          1       0.86      0.68      0.76       200
          5       0.93      0.97      0.95       822

avg / total       0.91      0.92      0.91      1022
```

Figure 7. Outcomes for binary classification

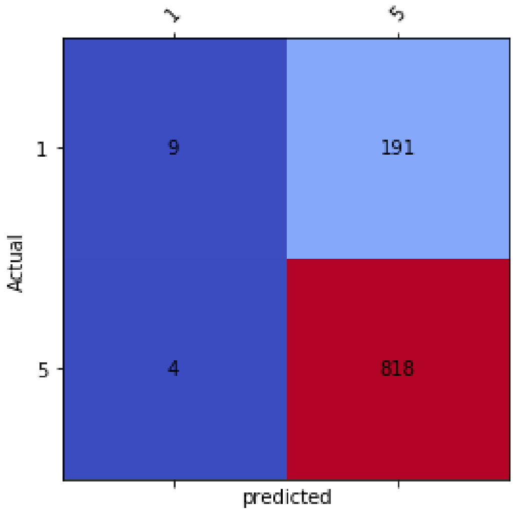
of Logistic Regression

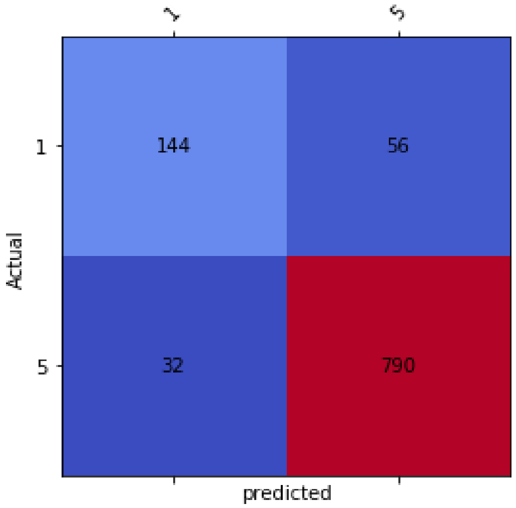


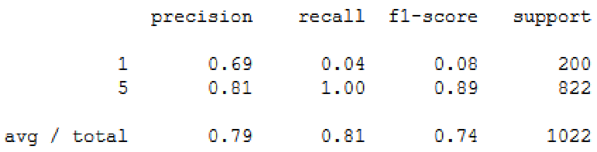
```
             precision    recall  f1-score   support

          1       0.78      0.48      0.59       200
          5       0.88      0.97      0.92       822

avg / total       0.86      0.87      0.86      1022
```

Figure 8. Outcomes for binary classification

of Random Forest

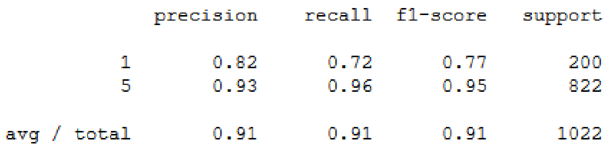






```
             precision    recall  f1-score   support

          1       0.69      0.04      0.08       200
          5       0.81      1.00      0.89       822

avg / total       0.79      0.81      0.74      1022
```

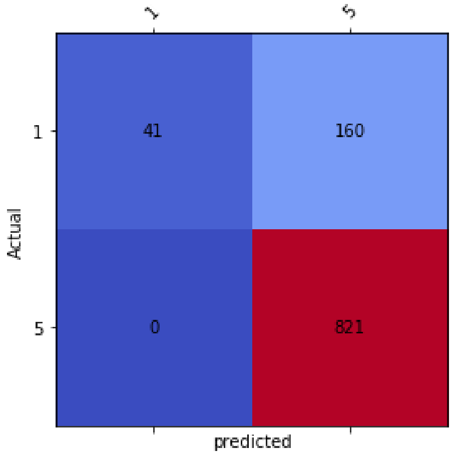
Figure 9. Outcomes for binary classification of K-Nearest Neighbors

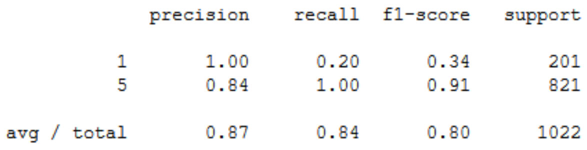


```
             precision    recall  f1-score   support

          1       0.82      0.72      0.77       200
          5       0.93      0.96      0.95       822

avg / total       0.91      0.91      0.91      1022
```

Figure 10. Outcomes for binary classification of Naïve Bayes

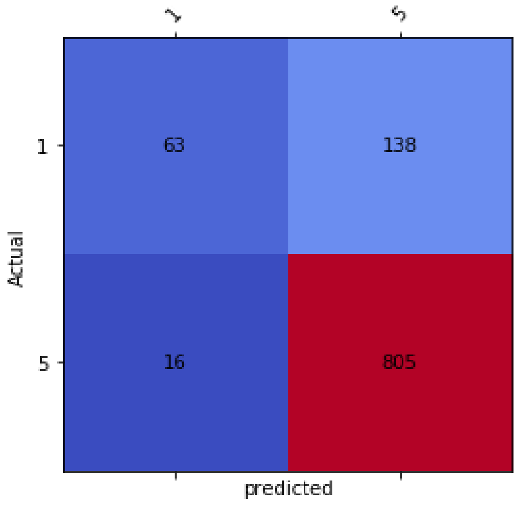
The star rating predictions using TF-IDF with Logistic Regression, Random Forest, and K-Nearest Neighbors are displayed in figures 11, 12, 13, and 14. The accuracy of all models is very similar, around 0.84, except for the naive Bayes model. However, their outcomes for binary classification differ significantly. It is evident that the precision, recall, and f1-score in the naive Bayes model are lower compared to those in the other models. Among the remaining models, it appears that the Logistic Regression model performs the best, as its precision score is the highest.

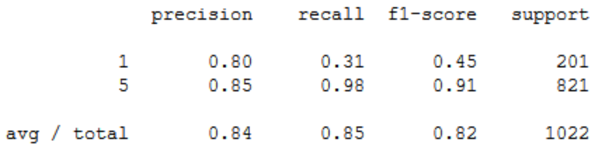




```
             precision    recall  f1-score   support

          1       1.00      0.20      0.34       201
          5       0.84      1.00      0.91       821

avg / total       0.87      0.84      0.80      1022
```

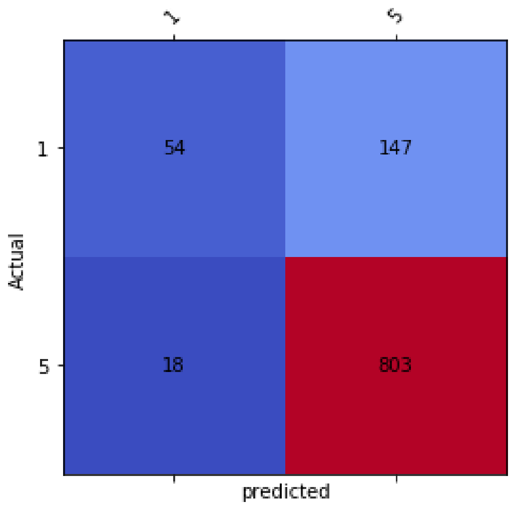
Figure 11. Outcomes for binary classification of Logistic Regression





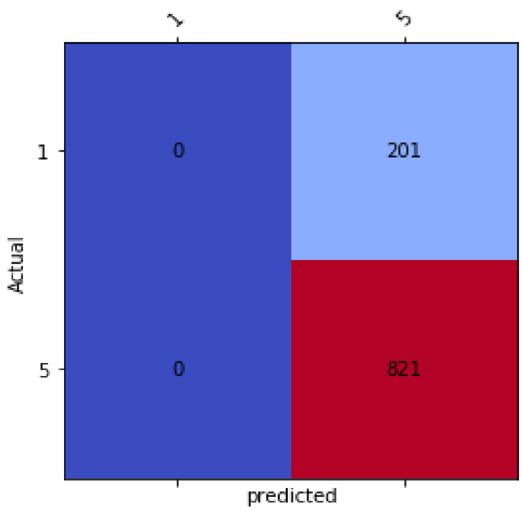
```
             precision    recall  f1-score   support

          1       0.80      0.31      0.45       201
          5       0.85      0.98      0.91       821

avg / total       0.84      0.85      0.82      1022
```

Figure 12. Outcomes for binary classification of Random Forest



```
             precision    recall  f1-score   support

          1       0.75      0.27      0.40       201
          5       0.85      0.98      0.91       821

avg / total       0.83      0.84      0.81      1022
```

Figure 13. Outcomes for binary classification of K-Nearest Neighbors



```
             precision    recall  f1-score   support

          1       0.00      0.00      0.00       201
          5       0.80      1.00      0.89       821

avg / total       0.65      0.80      0.72      1022
```

Figure 14. Outcomes for binary classification of Naïve Bayes

The ROC curves of TF-IDF and Bag-of-Words for Logistic Regression, Random Forest, and K-Nearest Neighbors are displayed in figures 11, 12, 13, and 14. Irrespective of the model applied, the TF-IDF-applied model holds greater value as its curve approaches closer to 1.0.

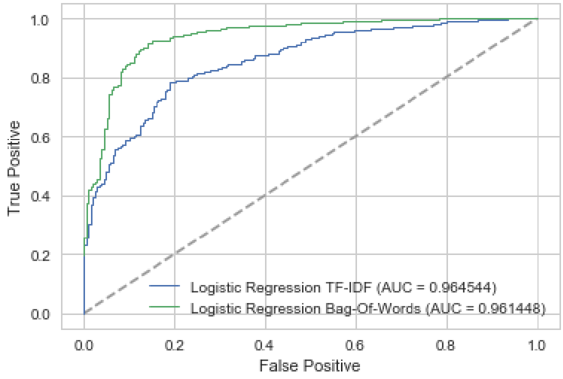


Figure 11. ROC for Logistic Regression

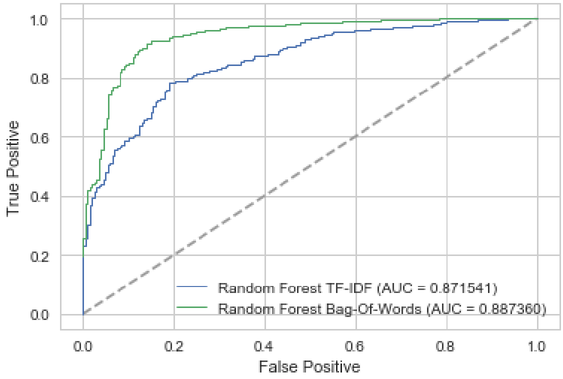


Figure 12. ROC for Random Forest

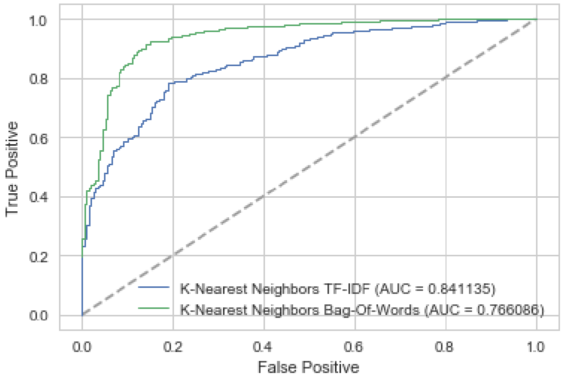


Figure 13. ROC for K-Nearest Neighbors

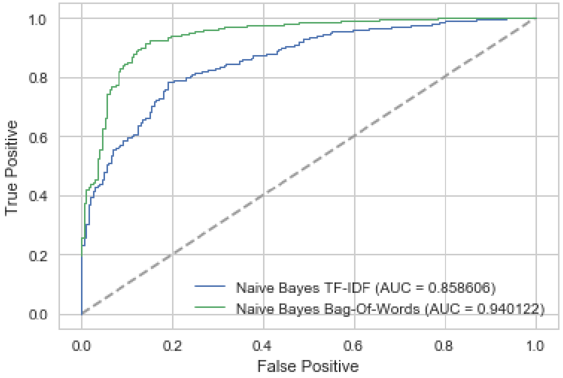


Figure 14. ROC for Naive Bayes

Conclusion

In this article, we analyzed Yelp datasets and employed machine learning models to predict the sentiment of reviews. We implemented four algorithmic models: logistic regression classifier, k-nearest neighbors, random forest, and naive Bayes classifier, and compared their performances in sentiment prediction. To enhance the prediction performance, we applied NLP techniques, including stemming the text and computing word occurrences or term frequency–inverse document frequency to transform the text notes. Finally, we evaluated and compared their performances using a test dataset. Based on the results from the test dataset, the optimized logistic regression trained with Tfidf-transformed data exhibited the best performance for sentiment prediction.

Conflict of interest

The authors declare no competing financial interest.

References

[1] K. Ravi, V. Ravi, A survey on opinion mining and sentiment analysis: tasks, approaches and applications, Knowledge-based systems 89 (2015) 14-46.

[2] S. Jiang, S. Chen, S. Chen, Novel antibacterial cotton textiles finished with siloxane sulfopropylbetaine, Fiber Soc. Spring (2011) 263-264.

[3] S. Chen, S. Chen, S. Jiang, M. Xiong, J. Luo, J. Tang, Z. Ge, Environmentally friendly antibacterial cotton textiles finished with siloxane sulfopropylbetaine, ACS applied materials & interfaces 3(4) (2011) 1154-1162.

[4] S. Chen, S. Chen, S. Jiang, Y. Mo, J. Tang, Z. Ge, Synthesis and characterization of siloxane sulfobetaine antimicrobial agents, Surface science 605(11-12) (2011) L25-L28.

[5] S. Chen, S. Chen, S. Jiang, Y. Mo, J. Luo, J. Tang, Z. Ge, Study of zwitterionic sulfopropylbetaine containing reactive siloxanes for application in antibacterial materials, Colloids and Surfaces B: Biointerfaces 85(2) (2011)

323-329.

[6] S. Jiang, Y. Liu, T. Wang, Y. Gu, Y. Luo, Design and Preparation of a Novel Antibacterial Hydrogel based on Maleimide-Thiol Conjugation, Journal ISSN 2766 (2023) 2276.

[7] S. Jiang, T. Zhang, Studies of a biocompatible maleimide-modified dextran and hyaluronic acid hydrogel system, Fine Chemical Engineering (2023) 100-109.

[8] S. Jiang, Y. Liu, T. Wang, Investigation of a Novel Biocompatible Fast-Gelling Hydrogel for Neurodegenerative Diseases, Biomedical Journal of Scientific & Technical Research 50(3) (2023) 1149-1157.

[9] S. JIANG, Y. LIU, Y. GU, SHORT PEPTIDE-BASED POLYSACCHARIDE HYDROGELS FOR TISSUE ENGINEERING: A MINI REVIEW.

[10] https://webcatalog.io/en/apps/yelp/

[11] S. Sawant, G. Pai, Yelp food recommendation system, SawantPai-YelpFoodRecommendationSystem. pdf (2013).

[12] A. Sihombing, A.C.M. Fong, Fake review detection on yelp dataset using classification techniques in machine learning, 2019 international conference on contemporary computing and informatics (IC3I), IEEE, 2019, pp. 64-68.

[13] S. Jiang, Y. Gu, E. Kumar, Stroke Risk Prediction Using Artificial Intelligence Techniques Through Electronic Health Records, Artificial Intelligence Evolution (2023) 88-98.

[14] https://www.kaggle.com/yelp-dataset/yelp-dataset

[15] J. Song, Y. Gu, E. Kumar, Chest disease image classification based on spectral clustering algorithm, Research Reports on Computer Science (2023) 77-90.

[16] Y. Gu, M. Wang, Y. Gong, X. Li, Z. Wang, Y. Wang, S. Jiang, D. Zhang, C. Li, Unveiling breast cancer risk profiles: a survival clustering analysis empowered by an online web application, Future Oncology (0) (2023).

[17] S. Jiang, Y. Gu, E. Kumar, Magnetic Resonance Imaging (MRI) Brain Tumor Image Classification Based on Five Machine Learning Algorithms, Cloud Computing and Data Science (2023) 122-133.