Indonesian Journal of Computer Science
ISSN 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam No. 1, Padang, Indonesia
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Knowledge Extraction of Gojek Application Review Using Aspect-based Sentiment Analysis

Anggun Ramadina[1], Ken Ditha Tania*, Ari Wedhasmara, Allsela Meiriza
anggun.ramadina123@gmail.com, kenyta.tania@gmail.com, a_wedhasmara@unsri.ac.id, allsela@unsri.ac.id
Sriwijaya University

Article Information

Submitted : 13 May 2024
Reviewed : 16 May 2024
Accepted : 15 Jun 2024

Keywords

Knowledge Extraction
Aspect-based sentiment analysis, Gojek
Support Vector Machine

Abstract

Social media is transforming business, costumer relations, and behavior. The involvement of customers and impacts from social media affect product or service performance. This is because the consumer's reviews are another important factor that builds brand image by showing product advantages and disadvantages. In order to enhance the business environment, the company may utilize this knowledge base to evaluate and enhance its business activities. The knowledge base approach might help the company or organization examine and extract information by using knowledge extraction. In this research, knowledge Extraction utilize Aspect-based Sentiment Analysis as the method for a better understanding and improving in analyze Gojek application reviews with Machine Learning approach by using Support Vector Machine (SVM) using the Kernel model with the result shows that there is no significant differences in each models. Futhermore, the knowledge extraction result using ABSA will be in XML format.

A. Introduction

The rising number of social media users has enormous impact on several industries, especially in terms of user decision-making. Sharing experiences on social media platforms encourages the communication of information, empowering users to make wise decisions. This heightened word-of-mouth has the ability to establish markets and enhance brand equity and company financial success. Gojek is a mobile application that accessible through Google Play Store or App Store that leading firm in Indonesia's creative economy industry has showcased its achievement by becoming the first Unicorn Company in 2016 and afterwards incorporating with Tokopedia in 2021 to establish GoTo. Gojek has consistently received the Brand Comparison award at the Top Brand Awards over the past five years, showcasing its dedication to quality and resilience[1] that would be feasible because Gojek really pay attention with their customer's feedback. Either customer's feedback on social media or comment and rating of the application. Nevertheless, the substantial and unrefined review data necessitates thorough processing in order to be utilised for sentiment analysis in order to comprehend and categorise opinions into several categories based on emotions and feelings.

The are two types of approaches used in conducting sentiment analysis, Machine Learning approach and Knowledge-based or lexicon-based[2], [3] Sentiment analysis itself is a field of study of analyzes people's opinions, judgements, attitudes, and emotions towards entities[2]. Several types of sentiment analysis able to conducted all at once to conduct a comprehensive study of the feedback, Saddam & Dewantara conduct SA for examinations of managing flood disaster in Jakarta[4], M. A. Jassim et al., purpose SA for new rating prediction of new films [5]. Knowledge base also aimed at serving business practices which primarily utilize by large organization or even individual that creates and consumes distributed knowledge. Knowledge is usually at a higher level of abstraction than a single item of a fact which can manually extracting evidence on behavior determinants related to specific types of behavior for specific social groups, although extremely laborintensive and challenging to collect and synthesize all knowledge[6], [7]. Provides heterogeneous information including both structured and unstructured data with different semanctics, knowledge base can help develop insight on problems which difficult to uncover [8], [9]. With Knowledge-based approach, we can utilize Knowledge Extraction (KE). KE is the process of extracting information and its relationship, generalizing the information and storing it in a structured manner in XML or Knowledge base format so that can be easily accessed and inferred. The extrated knowledge must be in machine-readable and machine intepretable format and must represent the knowledge in a way that facilitates inference. KE can use information extraction techniques which aim to extract (explicit) information with certain categories from a collection of documents[6], [10]. Since KE is aims to find entities, relations and event involving those entities from unstructured data and link them into existing knowledge bases, KE can be utilizing with Aspect Based Sentiment Analysis (ABSA).

ABSA which is one of the levels of sentiment analysis (SA) that has been considered the concept-level, focuses on the semantic analysis of the text throught the use of web ontologies and semantic networks[11], [12]. ABSA is paved the way to novel approaches for a better understanding, having process in different aspects like attributes, characteristics, or feature of product or service that provides benefits for a better aspect-aware text representation[13], [14], [15]. ABSA focusing on two tasks there are, Aspect Term Extraction (ATE) and Aspect Polarity Classification (APC). ATE work to identify different aspect mentioned in given sentence, refer to specific characteristic of product or service discussed in the feedback[16]. It is related to KE meaning which is the creation of knowledge from structured (relational databases, XML) and unstructured (text, documents, images) sources which contributes to establishing, improving, and affecting the knowledge that potentially applied in SA[6], [7], [17].

Previous studied that implied ABSA in KE is addresses automatic KE for ABSA in product review to introduce an approach to obtain a knowledge-based system to capture product aspects in specifi domain[18], in addition studies is about purpose incorporating multiple lexical knowledge sources into fine-tuning process of pre-trained transformer models of Targeted Aspect-based Financial Sentiment Analysis (TABFSA)[19]. According to the passage, the author studied KE using the ABSA method cause word identification in KE and ABSA are comparable and has relation in it. In this study, we utilize ABSA to construct KE which will be form in XML and open data so that can be reused for future research.

B. Research Method

The method of knowledge extraction of Gojek application reviews using ABSA will be used ABSA. The stages of this research can be seen in figure 1 below.

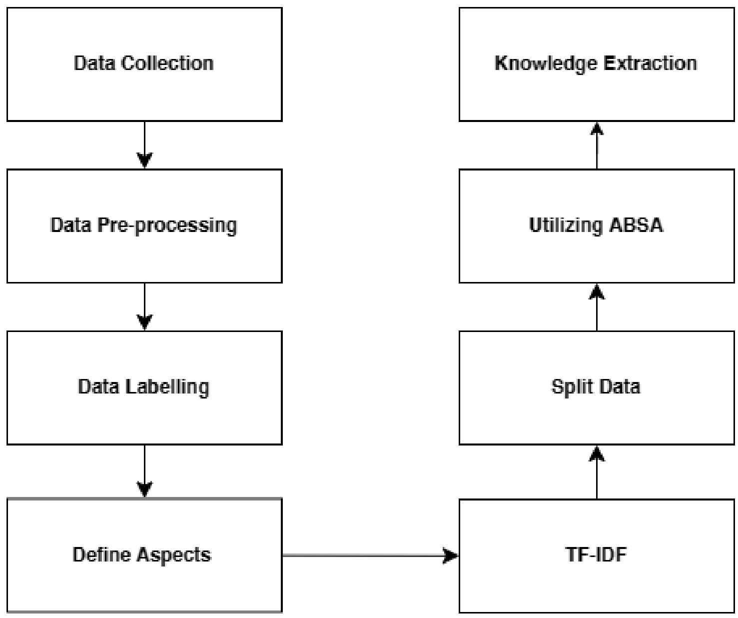

Figure 1. Research Framework

Data Collection

The very first stage of this reseach is data collection. Google Colaboratory uses Google Scrapper, one of the Python's libraries to collect data from Gojek application reviews on Google Play Store. The amount of the data is 400 which contains with the most recent feedback in Bahasa language.

Data Pre-processing

After collecting data, the collected data has to be separated from unstructured and duplicated data. The data was also cleared from null, normalization; process transforming data into a standardised format data, tokenizing; breaking down the structure of sentences into words and stopwords; removing words that have no potential or effect on the classification process; and stemming; elaborates senteces to find the basic word.

Data Labelling

In this step, the data will be divided into two parts: positive and negative. The rating with 4 and 5 was labeled as positive and ratings with 1, 2, and 3 were labelled negative. The purpose of labelling data used to train the system for the recognition of the pattern that is sought while testing the data once the result of the training has been caried out.

Define Aspects

Since this method use ABSA, we have to define the aspects that will be used to analyze. The researcher defines two aspects: Harga and Driver in Bahasa language whose contains with positive and negative words that can be seen on the table below.

Table 1. Aspects for Analysis

No	Aspect	Types	Words
1	Price	Positive	Cheap Affordable
		Negative	Expensive Unaccessible
2	Driver	Positive	Humble Kind Polite
		Negative	Rude Hostile Upset

Word Weighning

After defining the aspects, before we start the analysis we have to weigh the word using Term Frequency – Inverse Document Frequency (TF-IDF). TF-IDF is a method used to give a weigh for each words in data based on the its relevance.

Split Data

For this research, we need to split the data into data train and data test. This used for evaluate the prediction results with ratio 0.2 using scikit-learn to generate resulting 20% for data test and 80% for data train.

Utilizing ABSA

At this stage, the split data processed for modelling and analysing it using ABSA. This study use SVM models with linear, polynomial, and Radial Basis Function (RBF) kernels to fully analyze the aspects.

Cross Validation

Evaluate the model performance by dividing the dataset into the smallest subset. Training and testing data will be done alternatingly in every subset of data.

Table 2. Cross validation code program

<table>
<tr><th>No</th><th>Model</th><th>Code</th></tr>
<tr><td>1</td><td>Linear</td><td><pre>
[ ] #Cross Validation Using Linear Kernel - menghitung jarak antara dua titik, lebih sederhana
    ### create model
    clf_linear = SVC(kernel='linear')
    ### cross-validation evaluation
    cv_linear = cross_val_score(clf_linear, x_harga_train_vectorized, y_harga_train, scoring='accuracy', cv=5)
    ### Show CV result
    print("Accuracy Score Linear of : %0.4f" % (cv_linear.mean()))
</pre></td></tr>
<tr><td>2</td><td>Polynomial</td><td><pre>
[ ] #Cross Validation Using Polynomial Kernel
    clf_poly = SVC(kernel='poly')
    ### cross-validation evaluation
    cv_poly = cross_val_score(clf_poly, x_harga_train_vectorized, y_harga_train, scoring='accuracy', cv=5)
    ### Show CV result
    print("Accuracy Score Poly of : %0.4f" % (cv_poly.mean()))
</pre></td></tr>
<tr><td>3</td><td>RBF</td><td><pre>
[ ] #Cross Validation Using RBF Kernel
    clf_rbf = SVC(kernel='rbf')
    ### cross-validation evaluation
    cv_rbf = cross_val_score(clf_rbf, x_harga_train_vectorized, y_harga_train, scoring='accuracy', cv=5)
    ### show cv result
    print("Accuracy score linear of: %0.4f" % (cv_linear.mean))
</pre></td></tr>
</table>

Tuning Parameter with GridSearchCV

The optimization process of the model parameter is done by finding parameter combinations from the score list that have already been specified.

Table 3. Tuning parameter code program

<table>
<tr><th>No</th><th>Model</th><th>Code</th></tr>
<tr><td>1</td><td>Linear</td><td><pre>
[ ] #turning parameter with GridSearchCV
    #linear kernel
    param_grid = {'C': [0.001, 0.1, 1, 10, 100, 1000],
                  'kernel': ['linear']
                 }

    grid = GridSearchCV(svm.SVC(), param_grid, scoring='accuracy', refit = True, verbose = 3)
    # fitting the model for grid search
    grid.fit(x_harga_train_vectorized, y_harga_train)

    selected_kernel_linear = grid.best_params_['kernel']
    print("Selected Kernel: ", selected_kernel_linear)
    selected_C_linear = grid.best_params_['C']
    print("Selected C: ", selected_C_linear)

    best_params_linear_HT = grid.best_params_
    model_looks_linear_HT= grid.best_estimator_

    print(f'Best Score : {grid.best_score_}')
    print("best estimator: %s" % (model_looks_linear_HT))
</pre></td></tr>
<tr><td>2</td><td>Polynomial</td><td><pre>
[ ] #polynomial kernel
    param_grid = {'C': [0.001, 0.1, 1, 10, 100, 1000],
                  'kernel': ['poly'],
                  'degree':[1, 2, 3, 4, 5, 6],
                  'gamma' :['auto', 'scale']
                 }

    grid = GridSearchCV(svm.SVC(), param_grid, scoring='accuracy', refit = True, verbose = 3)
    # fitting the model for grid search
    grid.fit(x_harga_train_vectorized, y_harga_train)

    selected_kernel_poly = grid.best_params_['kernel']
    print("Selected Kernel: ", selected_kernel_poly)
    selected_C_poly= grid.best_params_['C']
    print("Selected C: ", selected_C_poly)
    selected_gamma_poly = grid.best_params_['gamma']
    print("Selected Gamma: ", selected_gamma_poly)
    selected_degree_poly = grid.best_params_['degree']
    print("Selected Degree: ", selected_degree_poly)

    best_params_poly_HT = grid.best_params_
    model_looks_poly_HT= grid.best_estimator_

    print(f'Best Score : {grid.best_score_}')
    print("best estimator: %s" % (model_looks_poly_HT))
</pre></td></tr>
</table>

<table>
<tr><td>3</td><td>RBF</td><td>#rbf kernel
param_grid = {'C': [0.001, 0.1, 1, 10, 100, 1000],
              'kernel': ['rbf'],
              'gamma' :['auto', 'scale']
              }

grid = GridSearchCV(svm.SVC(), param_grid, scoring='accuracy', refit = True, verbose = 3)
# fitting the model for grid search
grid.fit(x_harga_train_vectorized, y_harga_train)

selected_kernel_rbf = grid.best_params_['kernel']
print("Selected Kernel: ", selected_kernel_rbf)
selected_C_rbf = grid.best_params_['C']
print("Selected C: ", selected_C_rbf)
selected_gamma_rbf= grid.best_params_['gamma']
print("Selected Gamma: ", selected_gamma_rbf)

best_params_rbf_HT = grid.best_params_
model_looks_rbf_HT= grid.best_estimator_

print(f'Best Score : {grid.best_score_}')
print("best estimator: %s" % (model_looks_rbf_HT))</td></tr>
</table>

Cross Validation Tuned Parameter

In this process, the parameter model has already been optimized using the cross-validation method after tuning the parameter. This parameter is used to train the model with the dataset after optimization.

Table 4. Cross validation tuned parameter program code

<table>
<tr><th>No</th><th>Model</th><th>Code</th></tr>
<tr><td>1</td><td>Linear
Polynomial
RBF</td><td>#cv with Tuned Parameter
cross_val_score_linear= cross_val_score(model_looks_linear_HT, x_harga_train_vectorized, y_harga_train, scoring='accuracy', cv=5)
print("Accuracy Score Linear of : %0.4f" % cross_val_score_linear.mean(), cross_val_score_linear.std())

cross_val_score_poly = cross_val_score(model_looks_poly_HT, x_harga_train_vectorized, y_harga_train, scoring='accuracy', cv=5)
print("Accuracy Score poly of : %0.4f" % cross_val_score_poly.mean(), cross_val_score_poly.std())

cross_val_score_rbf = cross_val_score(model_looks_rbf_HT, x_harga_train_vectorized, y_harga_train, scoring='accuracy', cv=5)
print("Accuracy Score rbf of : %0.4f" % cross_val_score_rbf.mean(), cross_val_score_rbf.std())</td></tr>
</table>

C. Result and Discussion

Result for Price Aspect

Table 5. Result of cross validation

<table>
<tr><th>No</th><th>Model</th><th>Result</th></tr>
<tr><td>1</td><td>Linear</td><td>[11] print("Accuracy Score Linear of : %0.4f" % (cv_linear.mean()))

Accuracy Score Linear of : 0.8000</td></tr>
<tr><td>2</td><td>Polynomial</td><td>print("Accuracy Score Poly of : %0.4f" % (cv_poly.mean()))
Accuracy Score Poly of : 0.8000</td></tr>
<tr><td>3</td><td>RBF</td><td>print("Accuracy score linear of: %0.4f" % (cv_linear.mean()))
Accuracy score linear of: 0.8000</td></tr>
</table>

Table 6. Result of tuning parameter

<table>
<tr><th>No</th><th>Model</th><th>Result</th></tr>
<tr><td>1</td><td>Linear</td><td>Selected Kernel: linear
Selected C: 0.001
Best Score : 0.8
best estimator: SVC(C=0.001, kernel='linear')</td></tr>
<tr><td>2</td><td>Polynomial</td><td>Selected Kernel: poly
Selected C: 0.001
Selected Gamma: auto
Selected Degree: 1
Best Score : 0.8
best estimator: SVC(C=0.001, degree=1, gamma='auto', kernel='poly'</td></tr>
</table>

3	RBF	Selected Kernel: rbf Selected C: 0.001 Selected Gamma: auto Best Score : 0.8 best estimator: SVC(C=0.001, gamma='auto')

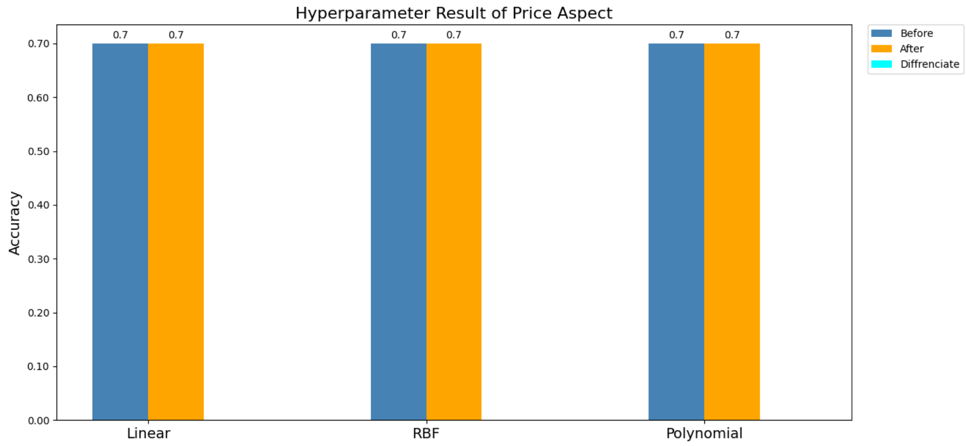

Figure 2. Hyperparameter result

Table 7. Result of cross validation tuned parameter (test dataset)

No	Model	Result
1	Linear	F1-SCORE 40.0 ACCURACY 66.66666666666666 PRECISION 33.33333333333333 RECALL 50.0 Confussion Matrix : [[2 0] [1 0]] Result : precision recall f1-score support -1.0 0.67 1.00 0.80 2 1.0 0.00 0.00 0.00 1 accuracy 0.67 3 macro avg 0.33 0.50 0.40 3 weighted avg 0.44 0.67 0.53 3
2	Polynomial	F1-SCORE 40.0 ACCURACY 66.66666666666666 PRECISION 33.33333333333333 RECALL 50.0 Confusion Matrix : [[2 0] [1 0]] Result : precision recall f1-score support -1.0 0.67 1.00 0.80 2 1.0 0.00 0.00 0.00 1 accuracy 0.67 3 macro avg 0.33 0.50 0.40 3 weighted avg 0.44 0.67 0.53 3

3	RBF	F1-SCORE 40.0 ACCURACY 66.66666666666666 PRECISION 33.33333333333333 RECALL 50.0 Confusion Matrix : [[2 0] [1 0]] Result : precision recall f1-score support -1.0 0.67 1.00 0.80 2 1.0 0.00 0.00 0.00 1 accuracy 0.67 3 macro avg 0.33 0.50 0.40 3 weighted avg 0.44 0.67 0.53 3

Table 8. Result of cross validation tuned parameter (train dataset)

No	Model	Result
1	Linear	F1-SCORE 25.641025641025646 ACCURACY 62.5 PRECISION 20.833333333333336 RECALL 33.33333333333333 Confusion Matrix : [[5 0 0] [1 0 0] [2 0 0]] Result : precision recall f1-score support -1.0 0.62 1.00 0.77 5 0.0 0.00 0.00 0.00 1 1.0 0.00 0.00 0.00 2 accuracy 0.62 8 macro avg 0.21 0.33 0.26 8 weighted avg 0.39 0.62 0.48 8
2	Polynomial	F1-SCORE 25.641025641025646 ACCURACY 62.5 PRECISION 20.833333333333336 RECALL 33.33333333333333 Confusion Matrix : [[5 0 0] [1 0 0] [2 0 0]] Result : precision recall f1-score support -1.0 0.62 1.00 0.77 5 0.0 0.00 0.00 0.00 1 1.0 0.00 0.00 0.00 2 accuracy 0.62 8 macro avg 0.21 0.33 0.26 8 weighted avg 0.39 0.62 0.48 8
3	RBF	F1-SCORE 25.641025641025646 ACCURACY 62.5 PRECISION 20.833333333333336 RECALL 33.33333333333333 Confusion Matrix : [[5 0 0] [1 0 0] [2 0 0]] Result : precision recall f1-score support -1.0 0.62 1.00 0.77 5 0.0 0.00 0.00 0.00 1 1.0 0.00 0.00 0.00 2 accuracy 0.62 8 macro avg 0.21 0.33 0.26 8 weighted avg 0.39 0.62 0.48 8

Result for Aspect Driver

Table 9. Result of cross validation

No	Model	Result
1	Linear	`print("Accuracy Score Linear of : %0.4f" % (cv_linear.mean()))` `Accuracy Score Linear of : 0.4000`
2	Polynomial	`print("Accuracy Score Poly of : %0.4f" % (cv_poly.mean()))` `Accuracy Score Poly of : 0.6000`
3	RBF	`print("Accuracy score linear of: %0.4f" % (cv_linear.mean()))` `Accuracy score linear of: 0.8000`

Table 10. Result of tuning parameter

No	Model	Result

1	Linear	`Selected Kernel:  linear`
		`Selected C:  0.001`
		`Best Score : 0.6`
		`best estimator: SVC(C=0.001, kernel='linear')`

2	Polynomial	`Selected Kernel:  poly`
		`Selected C:  0.001`
		`Selected Gamma:  auto`
		`Selected Degree:  1`
		`Best Score : 0.6`
		`best estimator: SVC(C=0.001, degree=1, gamma='auto', kernel='poly')`

3	RBF	`Selected Kernel:  rbf`
		`Selected C:  0.001`
		`Selected Gamma:  auto`
		`Best Score : 0.6`
		`best estimator: SVC(C=0.001, gamma='auto')`

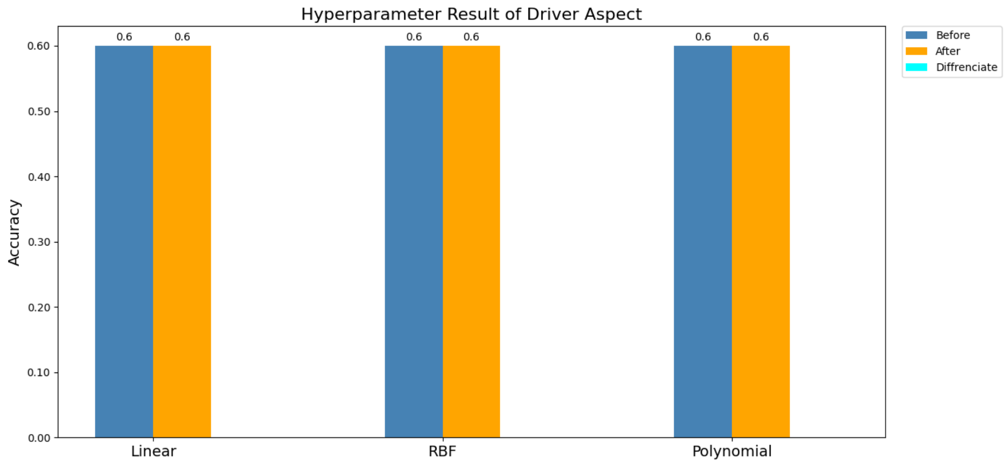

Figure 3. Hyperparameter result

Table 11. Result of cross validation tuned parameter (test dataset)

No	Model	Result
1	Linear	F1-SCORE 25.0 ACCURACY 33.33333333333333 PRECISION 16.666666666666664 RECALL 50.0 Confussion Matrix : [[0 2] [0 1]] Result : precision recall f1-score support -1.0 0.00 0.00 0.00 2 1.0 0.33 1.00 0.50 1 accuracy 0.33 3 macro avg 0.17 0.50 0.25 3 weighted avg 0.11 0.33 0.17 3
2	Polynomial	F1-SCORE 25.0 ACCURACY 33.33333333333333 PRECISION 16.666666666666664 RECALL 50.0 Confusion Matrix : [[0 2] [0 1]] Result : precision recall f1-score support -1.0 0.00 0.00 0.00 2 1.0 0.33 1.00 0.50 1 accuracy 0.33 3 macro avg 0.17 0.50 0.25 3 weighted avg 0.11 0.33 0.17 3
3	RBF	F1-SCORE 25.0 ACCURACY 33.33333333333333 PRECISION 16.666666666666664 RECALL 50.0 Confusion Matrix : [[0 2] [0 1]] Result : precision recall f1-score support -1.0 0.00 0.00 0.00 2 1.0 0.33 1.00 0.50 1 accuracy 0.33 3 macro avg 0.17 0.50 0.25 3 weighted avg 0.11 0.33 0.17 3

Table 12. Result of cross validation tuned parameter (train dataset)

No	Model	Result
1	Linear	F1-SCORE 23.80952380952381 ACCURACY 55.55555555555556 PRECISION 18.51851851851852 RECALL 33.33333333333333 Confusion Matrix : [[0 0 3] [0 0 1] [0 0 5]] Result : precision recall f1-score support -1.0 0.00 0.00 0.00 3 0.0 0.00 0.00 0.00 1 1.0 0.56 1.00 0.71 5 accuracy 0.56 9 macro avg 0.19 0.33 0.24 9 weighted avg 0.31 0.56 0.40 9

2	Polynomial	F1-SCORE 23.80952380952381 ACCURACY 55.55555555555556 PRECISION 18.51851851851852 RECALL 33.33333333333333 Confusion Matrix : [[0 0 3] [0 0 1] [0 0 5]] Result : precision recall f1-score support -1.0 0.00 0.00 0.00 3 0.0 0.00 0.00 0.00 1 1.0 0.56 1.00 0.71 5 accuracy 0.56 9 macro avg 0.19 0.33 0.24 9 weighted avg 0.31 0.56 0.40 9
3	RBF	F1-SCORE 23.80952380952381 ACCURACY 55.55555555555556 PRECISION 18.51851851851852 RECALL 33.33333333333333 Confusion Matrix : [[0 0 3] [0 0 1] [0 0 5]] Result : precision recall f1-score support -1.0 0.00 0.00 0.00 3 0.0 0.00 0.00 0.00 1 1.0 0.56 1.00 0.71 5 accuracy 0.56 9 macro avg 0.19 0.33 0.24 9 weighted avg 0.31 0.56 0.40 9

After seeing the classification result against the models, we extracting the information into uncover a new knowledge. KE have emerged as a powerful approach across various fields, facilitating automatic acquisition and representation of valuable insight from each sample[20] with process involves classifying the extracted information to ensure its generality, accessibility, readability, and machine interpretation. As the result of extraction knowledge from ABSA we can conclude if the sentiment of Gojek application reviews for this past year decreased mainly on driver aspect. Due to drivers characteristics, the majority of consumers complaints about the unruly behaviour of Gojek's Driver. Followed by the pricing adjustement that increases. Knowledge should be shared between members in organization and between organizations. Nevertheless, the KE of this analysis will be used by the company in considering the business goals and supporting in company's business planning.

KE that has been generated can be storing using XML format. In recent years, storing knowledges in XML format has gained much popularity and has lead the interest to storing of large data repositories in XML format. The flexibility and expressive nature of XML allows to organize knowledge in textual contents into hierarchical structures and a standard model to store and transport data[21].

Afterwards, the outcomes of the ABSA which were transformed into KE and recorded in XML format possibly seen as this follows.

```xml
<document>
<title> Knowledge Extraction of Gojek Application Review Using Aspect-based Sentiment Analysis</title>
    <content>
        <aspect ="Harga">
                <positive_words> murah, terjangkau </positive_words>
                <negative_words>mahal, tidak terjangkau</negative_words>
                    <positive_words_dataset>
                        Cepat, aman, hemat
                        Diskon, drivernya, ramah
                        Pilih murah nyaman
                    </positive_words_dataset>
                    <negative_words_dataset>
                        Bantu harga lumayan mahal banding pakai aplikasi ojol yang satu
                        Mahal sesuai
                        Lot mahal kecewa
                        Gak mahal
                        Gojek ribet klaim diskon biaya kirim mahal harga dri makan beli
                        Kesini lot gak sat set layan ongkir mahal
                        Susah banget ngecek harga pickup ga muncul
                        Mahal ongkirnya harga makan gilak
                    </negative_words_dataset>
                 <absa_result> <!--Linear, Polynomial, and RBF-->
                    <cross_validation>
                        <accuracy>80%</accuracy>
                    </cross_validation>
                    <tuning_parameter>
                        <score>0.8</score>
                        <C>0.001</C>
                        <Gamma>auto</Gamma>
                    </tuning_parameter>
                    <cross_validation_tuned_parameter_test>
                        <F1_score>0.25%</F1_score>
                        <accuracy>33.33%</accuracy>
                        <precision>16.6%</precision>
                        <recall>50%</recall>
                        <confussion_matrix>
                            <true_negative>1</true_negative>
                            <false_positive>0</false_positive>
                            <false_negative>2</false_negative>
                            <true_positive>0</true_positive>
                        </confussion_matrix>
                    </cross_validation_tuned_parameter_test>
                    <cross_validation_tuned_parameter_train>
                        <F1_score>28.57%</F1_score>
                        <accuracy>75%</accuracy>
                        <precision>25%</precision>
                        <recall>33.33%</recall>
                        <confussion_matrix>
                        <true_negative>6</true_negative>
                            <false_positive>0</false_positive>
                            <false_negative>2</false_negative>
                            <true_positive>0</true_positive>
                        </confussion_matrix>
                    </cross_validation_tuned_parameter_train>
                </absa_result>
        </aspect = "Harga">
        <aspect = "Driver">
            <positive_words>ramah, baik, sopan</positive_words>
            <negative_words>tidak sopan, tidak ramah, marah</negative_words>
                <positive_words_dataset>
                    sopir ramah
                    diskon drivernya ramah ramah
                    drivernya bersih cepat ramah
                    dpt driver yg baik
                </positive_words_dataset>
                <negative_words_dataset>
                    drivernya banyak yg kurang ramah ada yang bahkan sampe marah kalo kayak gitu pelangganan jadi tidak nyaman tolong yg baik
                    mengganggu gak sopan dan tidak beretika
                    drivernya suka gak sopan
                    driver sinting giliran udah deket dicancel
                </negative_words_dataset>
            <absa_result> <!--Linear, Polynomial, and RBF-->
                    <cross_validation>
                        <accuracy="linear">0.4%</accuracy="linear">
                        <accuracy="polynomial">0.6%</accuracy="polynomial">
                        <accuracy="rbf">0.8%</accuracy="rbf">
                    </cross_validation>
                    <tuning_parameter>
                        <score>0.6</score>
                        <C>0.001</C>
                        <Gamma>auto</Gamma>
                    </tuning_parameter>
```

```xml
        <cross_validation_tuned_parameter_test>
          <F1_score>25%</F1_score>
          <accuracy>33.33%</accuracy>
          <precision>16.6%</precision>
          <recall>50%</recall>
          <confussion_matrix>
            <true_negative>0</true_negative>
            <false_positive>2</false_positive>
            <false_negative>0</false_negative>
            <true_positive>1</true_positive>
          </confussion_matrix>
        </cross_validation_tuned_parameter_test>
        <cross_validation_tuned_parameter_train>
          <F1_score>28.80%</F1_score>
          <accuracy>55%</accuracy>
          <precision>18.51%</precision>
          <recall>33.33%</recall>
          <confussion_matrix>
            <true_negative>0</true_negative>
            <false_positive>0</false_positive>
            <false_negative>1</false_negative>
            <true_positive>5</true_positive>
          </confussion_matrix>
        </cross_validation_tuned_parameter_train>
      </absa_result>
    </aspect = "Driver">
  </content>
</document>
```


Figure 4. Knowledge extraction from ABSA Gojek reviews in XML format

D. Conclusion

This research present information regarding the outcomes of ABSA on the Gojek application reviews. Performing SVM with Kernel models, linear, polynomial, and RBF resulting no significant differences for Harga aspect analysis. Instead, driver aspect show differences in linear models for cross validation and parameter tuning result. This may occur due to a lack of sufficient data during the processing phase. The KE in this research also stored in an XML format which expected to be used by companies or other researchers to obtain information of the performance Gojek application toward user's reviews. Expected for futher research to explore alternative methods of analysis to achive optimal and superior outcomes.

E. References

[1] I. Th and A. Revan, "Sentiment Analysis for Customer Review: Case Study of GO-JEK Expansion," *Journal of Information Systems Engineering and Business Intelligence*, vol. 6, pp. 1–8, Apr. 2020, doi: 10.20473/jisebi.6.1.1-8.

[2] B. Liu, *Sentiment Analysis and Opinion Mining*. Cham: Springer International Publishing, 2012. doi: 10.1007/978-3-031-02145-9.

[3] G. D'Aniello, M. Gaeta, and I. La Rocca, "KnowMIS-ABSA: an overview and a reference model for applications of sentiment analysis and aspect-based sentiment analysis," *Artif Intell Rev*, vol. 55, no. 7, pp. 5543–5574, Oct. 2022, doi: 10.1007/s10462-021-10134-9.

[4] M. A. Saddam, E. K. Dewantara, and A. Solichin, "Sentiment Analysis of Flood Disaster Management in Jakarta on Twitter Using Support Vector Machines," *Sinkron*, vol. 8, no. 1, pp. 470–479, Jan. 2023, doi: 10.33395/sinkron.v8i1.12063.

[5] M. A. Jassim, D. H. Abd, and M. N. Omri, "Machine learning-based new approach to films review," *Soc Netw Anal Min*, vol. 13, no. 1, p. 40, Mar. 2023, doi: 10.1007/s13278-023-01042-7.

[6] T. S. Sonnenschein, G. A. de Wit, N. R. den Braver, R. C. H. Vermeulen, and S. Scheider, “Validating and constructing behavioral models for simulation and projection using automated knowledge extraction,” *Inf Sci (N Y)*, vol. 662, p. 120232, Mar. 2024, doi: 10.1016/j.ins.2024.120232.
[7] G. K. Palshikar *et al.*, “RINX: A system for information and knowledge extraction from resumes,” *Data Knowl Eng*, vol. 147, p. 102202, Sep. 2023, doi: 10.1016/j.datak.2023.102202.
[8] M. Li *et al.*, “GAIA: A Fine-grained Multimedia Knowledge Extraction System,” in *Annual Meeting of the Association for Computational Linguistics*, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:218551030
[9] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma, “Collaborative Knowledge Base Embedding for Recommender Systems,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM, Aug. 2016, pp. 353–362. doi: 10.1145/2939672.2939673.
[10] Y. Chasseray, A.-M. Barthe-Delanoë, S. Négny, and J.-M. Le Lann, “Knowledge extraction from textual data and performance evaluation in an unsupervised context,” *Inf Sci (N Y)*, vol. 629, pp. 324–343, Jun. 2023, doi: 10.1016/j.ins.2023.01.150.
[11] S. Poria, E. Cambria, G. Winterstein, and G.-B. Huang, “Sentic patterns: Dependency-based rules for concept-level sentiment analysis,” *Knowl Based Syst*, vol. 69, pp. 45–63, Oct. 2014, doi: 10.1016/j.knosys.2014.05.005.
[12] O. Alqaryouti, N. Siyam, A. Abdel Monem, and K. Shaalan, “Aspect-based sentiment analysis using smart government review data,” *Applied Computing and Informatics*, vol. 20, no. 1/2, pp. 142–161, Jan. 2024, doi: 10.1016/j.aci.2019.11.003.
[13] M. Z. Asghar, F. M. Kundi, S. Ahmad, A. Khan, and F. Khan, “T-SAF: Twitter sentiment analysis framework using a hybrid classification scheme,” *Expert Syst*, vol. 35, no. 1, Feb. 2018, doi: 10.1111/exsy.12233.
[14] S. Y. Ganeshbhai and B. K. Shah, “Feature based opinion mining: A survey,” in *2015 IEEE International Advance Computing Conference (IACC)*, IEEE, Jun. 2015, pp. 919–923. doi: 10.1109/IADCC.2015.7154839.
[15] M. S. Akhtar, D. Gupta, A. Ekbal, and P. Bhattacharyya, “Feature selection and ensemble construction: A two-step method for aspect based sentiment analysis,” *Knowl Based Syst*, vol. 125, pp. 116–135, Jun. 2017, doi: 10.1016/j.knosys.2017.03.020.
[16] G. Zhao, Y. Luo, Q. Chen, and X. Qian, “Aspect-based sentiment analysis via multitask learning for online reviews,” *Knowl Based Syst*, vol. 264, p. 110326, Mar. 2023, doi: 10.1016/j.knosys.2023.110326.
[17] J. Unbehauen, S. Hellmann, S. Auer, and C. Stadler, “Knowledge Extraction from Structured Sources,” 2012, pp. 34–52. doi: 10.1007/978-3-642-34213-4_3.
[18] A.-D. Vo, Q.-P. Nguyen, and C.-Y. Ock, “Automatic Knowledge Extraction for Aspect-based Sentiment Analysis of Customer Reviews,” in *Proceedings of the 10th International Conference on Computer Modeling and Simulation*, New York, NY, USA: ACM, Jan. 2018, pp. 110–113. doi: 10.1145/3177457.3177462.

[19] K. Du, F. Xing, and E. Cambria, "Incorporating Multiple Knowledge Sources for Targeted Aspect-based Financial Sentiment Analysis," *ACM Trans Manag Inf Syst*, vol. 14, no. 3, pp. 1–24, Sep. 2023, doi: 10.1145/3580480.
[20] S. Fu, T. Dong, Z. Wang, and Y. Tian, "Weakly privileged learning with knowledge extraction," *Pattern Recognit*, vol. 153, p. 110517, Sep. 2024, doi: 10.1016/j.patcog.2024.110517.
[21] N. T. Thasleena and S. C. Varghese, "Enhanced Associative Classification of XML Documents Supported by Semantic Concepts," *Procedia Comput Sci*, vol. 46, pp. 194–201, 2015, doi: 10.1016/j.procs.2015.02.011.