Accredited Ranking SINTA 2
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online on: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 6 No. 4 (2022) 591 – 598 ISSN Media Electronic: 2580-0760

Aspect Based Sentiment Analysis with FastText Feature Expansion and Support Vector Machine Method on Twitter

Muhammad Afif Raihan[1], Erwin Budi Setiawan[2]
[1,2]Informatics, School of Computing, Telkom University
[1]mafifraihan@student.telkomuniversity.ac.id, [2]erwinbudisetiawan@telkomuniversity.ac.id

Abstract

Social media such as Twitter has now become very close to society. Twitter users can express current issues, their opinions, product reviews, and many other things both positive and negative. Twitter is also used by companies to monitor the assessment of their products among the public as insight that will be used to evaluate what aspects of their products need to be further developed. Twitter with its limitation of only allowing users to post a maximum tweet of 280 characters will make a lot of abbreviated and difficult to understand words used, so it will allow vocabulary mismatch problems to occur. Therefore, in this paper, research conducted on aspect-based sentiment analysis of Telkomsel's products from the aspects of signal and service by applying feature expansion using Fasttext word embedding to overcome vocabulary mismatch problem and classification with the Support Vector Machine (SVM) method. Sampling technique with Synthetic Minority Oversampling Technique (SMOTE) used to overcome data imbalance. The experimental results show that feature expansion can increase the performance of model. The final results obtained F1-Score value of the model for the signal aspect increased by 27.91% with F1-Score 95.93%, and for the service aspect increased by 42.36% with F1-Score 94.53%.

Keywords: twitter, aspect-based sentiment analysis, feature expansion, fasttext, svm

1. Introduction

In this era, social media has become almost inseparable from people's lives. One of the most frequently used social media is Twitter. Twitter allows users to share with other users by posting a tweet that can contain photos, videos, links, and hashtags[1]. Twitter users usually post tweets related to their thoughts, their opinions, current issues, and make good or bad comments about a product. Everything in the tweet above can be referred to as sentiment. In posting tweets on Twitter, users are limited to a maximum of 280 characters per tweet. Therefore, Tweets are often shortened and very difficult to understand due to the varied use of words and emoticons, thus causing vocabulary mismatches may occur[2].

Sentiment analysis has attracted attention in recent decades in research in the field of Natural Language Processing (NLP)[3]. Sentiment analysis is a process to identify or categorize user opinions in the form of text on anything such as movies, products, events, and other things whether they include positive, negative, or neutral sentiments[4]. Sentiment analysis can be very useful for the task of identifying, extracting, and learning subjective information about, for example, a company's product[5]. Sometimes, the companies are interested in more detailed insights into the sentiments expressed towards their products such as what aspects of their products need to be evaluated. In the Advancement of sentiment analysis, today is currently encountered as a new branch of sentiment analysis, namely aspect-based sentiment analysis.

Nipuna Eka Panala et al.[6], conducted an experiment on aspect-based sentiment analysis of laptop reviews using a supervised learning approach. The methods used are SVM and logistic regression. The results obtained are SVM getting better accuracy with a score of 73.18% while logistic regression is 72.39%. Irbah and Sibaroni in their research[7], discuss about aspect-based sentiment analysis of beauty product reviews in the Female Daily community using SVM classification. There are 3 aspects used, namely price, packaging, and aroma. The results of the accuracy score from this study are 86% for the aroma aspect, 92% for the packaging aspect, and 93% for the price aspect. Iskandar and Nataliani in their research[8], conducted sentiment analysis research based on Youtube comments related to the Samsung Galaxy Z Flip 3 Gadget. The aspects taken in this study are design aspect, price aspect,

Accepted: 28-06-2022 | Received in revised: 15-07-2022 | Published: 22-08-2022

specification aspect, and brand image aspect. In this study, SMOTE technique for sampling was used to overcome data imbalance, and SVM, NB, and k-NN for the classification methods. The results showed that SVM showed the best results with an accuracy score of 97.44% for the price aspect, 96.22% for the specification aspect, 94.40% for the design aspect, and 97.63% for the brand image aspect.

In research[2], Erwin B. Setiawan et al. conducted experiments related to topic classification in a tweet using SVM, Naïve Baiyes, and Logistic Regression and implemented feature expansion with Word2Vec word embedding using IndoNews and GoogleNews datasets. The results obtained show that feature expansion using Word2Vec word embedding with GoogleNews dataset can consistently improve performance. In Logistic Regression, the largest accuracy result is 58.86%, then SVM by 54.67% and Naïve Baiyes by 52.99%. In study[9], Ibrahim Kaibi et al. conducted experiments to compare 3 word embedding models (Glove, Word2Vec, Fasttext) which were then combined with 6 machine learning algorithms namely Gaussian Naïve Bayes, Linear SVC, NuSVC, Logistic Regression, SGD, and Random Forest on Arabic Twitter datasets. From this experiment, it is found that all algorithms combined with Fasttext word embedding get superior results than when combined with Glove or Word2Vec word embedding. The best results are obtained from the combination between Fasttext and nuSVC with a Precision value of 84.89%, Recall 84.29%, and F1-Score 83.85%.

Based on several previous researches above, it shown that many researches have used SVM classification method for text classification research such as sentiment analysis. The Implementation of feature expansion can help overcome vocabulary mismatches problem and improve the performance of classification model[2]. But, there is no research that uses feature expansion and SVM for aspect-based sentiment analysis. Therefore, this paper aims to analyze the impact of implementing feature expansion using Fasttext word embedding and SVM classification for aspect-based sentiment analysis on Twitter about Telkomsel's products. Telkomsel was chosen because Telkomsel is the cellular operator with the most extensive coverage and the most users in Indonesia. Therefore, it is expected that there will be many reviews from customers regarding Telkomsel's products. Due to the data used is imbalanced, data sampling will be carried out using two techniques, that is undersampling technique and SMOTE technique.

The rest of the paper is organized as follows. Section 2 describes the system that is built for feature expansion using Fasttext word embedding and SVM classifier for aspect-based sentiment analysis. Section 3 provides the result and discussion of the experiments in this research. section 4 conclusion of the experiments.

2. Research Methods

The system plan of the aspect-based sentiment analysis system using Fasttext feature expansion and SVM is shown in Figure 1. This research start with collecting data from Twitter, and then continued with labelling, pre-processing, feature extraction with TF-IDF, implementing the feature expansion using Fasttext word embedding, classification process with SVM, and lastly evaluating the model.

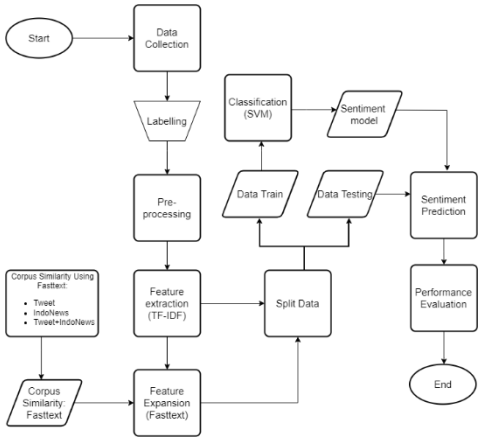

Figure 1. Aspect Based Sentiment Analysis with Fasttext and SVM

2.1 Data Collection

In this paper, the data used is tweet data from twitter. The tweet data was collected using the SNScrape library in the python programming language. The data that we collect is data related to Telkomsel, a cellular company in Indonesia. In collecting data several keywords related to Telkomsel's products in the signal and service aspects were used as listed in the Table 1. 16.978 tweets were successfully collected in the data collecting process. The data used to help the feature expansion process is Indonews Corpus, it is Indonesian article data from several media in Indonesia such as Tempo, Liputan 6, CNN Indonesia, Kompas, Detik.com, and Republika. The total number of articles owned is 142.545 articles.

Table 1. Keywords Related to Telkomsel's Products

No.	Keyword
1	sinyal telkomsel
2	jaringan telkomsel
3	telkomsel lemot
4	kualitas telkomsel
5	telkomsel down
6	telkomsel roaming
7	telkomsel gangguan
8	telkomsel error
9	cs telkomsel
10	grapari
11	pelayanan telkomsel
12	mytelkomsel
13	pelayanan sms telkomsel

DOI: https://doi.org/10.29207/resti.v6i4.4187

2.2 Data Labelling

The collected tweet data will be labeled to help the classification process. Data labeling is done manually by a team of 7 persons by dividing 1 tweet labeled by 3 persons and the most votes are taken as the tweet label. Data labelling is carried out on each tweet based on signal aspect and service aspect by labeling 1 if the tweet discussing that aspect is positive, label -1 if the tweet discussing that aspect is negative, and will be labeled 0 if the tweet discussing that aspect is neutral or does not discuss the aspect. Table 2 shows an example of tweet labelling.

Table 2. Example of Tweet Labelling

Tweet	Signal	Service
Ini sinyal telkomsel knpa jelek bgt y skrg	-1	0
@jessptra pake telkomsel biar bagus sinyal nya	1	0
@milkeesway @Telkomsel CS Telkomsel emang selalu responsif. Sabar aja nanti pasti dibales	0	1

After all tweet data is labeled, the distribution of label data for each aspect is obtained as shown in Table 3. There is an imbalance of data in all aspects. On the signal aspect, the positive label is too low compared to other labels, while on the service aspect, the neutral label is too high compared to other labels.

Table 3. Data Label Distribution of Each Aspect

Aspect	Positive	Negative	Neutral
Signal	471	9649	6870
Service	1216	2225	13546

2.3 Pre-processing

Crawling tweet data will definitely be found many tweets that contain noise and misspelling. Therefore, it is necessary to do preprocessing. The purpose of pre-processing is to remove noise, misspelling in tweets as well as to make the data more machine readable to reduce ambiguity in feature extraction[10]. Python libraries such as re, string, Sastrawi, and Natural Language Toolkit (NLTK) are used to help in pre-processing process.

Pre-processing is divided into 5 steps that consist of case folding, data cleaning, word normalization, stopwords removal, and stemming. The first step is case folding which is the stage of converting all words in the sentences into lowercase letters. The second step is data cleaning which is the process of removing urls, hashtags, username tags, numbers, symbols, and punctuation from the tweet. The third step is word normalizatoin which is the stage of converting slang words, abbreviated words, typos, informal words into correct and formal words with the help of a manually created dictionary [11]. The fourth step is stopwords removal which is the stage of removing words that have no meaning, and if the word is removed it does not affect the meaning of the sentence such as "dan", "atau", "tetap", "ini", "itu", "adalah", and others[2]. The last step is stemming which is a stage to change a word into its base word by removing prefixes, infixes, suffixes, and confixes (combination of prefix and suffix)[2].

2.4 Term Frequency – Inverse Document Frequency (TF-IDF)

TF-IDF is a machine learning method for NLP that reflects how important words or documents are in a corpus[12]. This method is used to calculate the weight of a word in a tweet that is efficient, easy, and produces accurate results[2]. In this paper, TF-IDF is used for feature extraction. The calculation for the weigth of word w in a tweet T using TF-IDF is defined as in formula (1).

$$W_{wi} = tf_{wT} * log(\frac{N_T}{n_w}) \quad (1)$$

Where tf_{wT} is the number of occurrences of word w in tweet T, N is the number of tweets we have, and n_w is the number of tweets containing word w [2].

2.5 Fasttext

Fasttext is a word embedding project that is part of Facebook's open sourced research[13]. Fasttext is a fast and effective method for learning word representations and performing text classification[13]. Fasttext learns words by considering the subwords of the word, then each word will be represented as a set of n-gram characters which allows Fasttext to capture the meaning of shorter words and allows embedding to understand the suffixes and prefixes of words[14]. Because of Fasttext considers subwords, it will be able to capture words that rarely appear in documents and handle the problem of unrecognized words, also known as Out of Vocabulary words (OOV). In this paper, Fasttext is used to create a similarity corpus using tweets data, IndoNews data, and a combination of tweets and IndoNews data.

2.6. Feature Expansion

As stated earlier, this paper implementing feature expansion using Fasttext word embedding to solve the vocabulary mismatch problem. The basic idea of using word embedding is to identify missing words in the tweet representation if they can be replaced with semantically meaningful words[2]. In this paper, feature expansion is done by utilizing similarity corpus that has been previously created using Fasttext to obtain related word. It takes a word as input and gives an output of a set of semantically related words and also their similarity values[2]. An example for the result of similarity corpus, given a word "telkomsel". In corpus IndoNews with Top 10 features the result of similarity corpus is as shown in Table 4 that shows 10 words that have similarities with the word "telkomsel". Whereas the Rank-1 to Rank-10 columns express the degree of proximity of the word that similar to the word

DOI: https://doi.org/10.29207/resti.v6i4.4187

"telkomsel". Rank-1 indicates the similar word with highest degree of proximity, and Rank-10 indicates the similar word with the lowest degree of proximity in the similarity corpus.

Table 4. Top 10 Similar Words of Word "telkomsel"

Word	Rank	Similar Word
Koneksi	Rank-1	telkomseltelkomsel
	Rank-2	mytelkomsel
	Rank-3	telkomselfest
	Rank-4	tekomsel
	Rank-5	telko
	Rank-6	ririek
	Rank-7	telkomunikasi
	Rank-8	telkom
	Rank-9	trikomsel
	Rank-10	telkomgroup

For the process of feature expansion, this paper uses the same method as that done by Erwin et al.[2]. As an example, given a tweet "... aplikasi mytelkomsel gangguan". Suppose "telkomsel" is a word whose feature value in the tweet representation is zero. Suppose also that the similar words of "telkomsel" as returned by Fasttext are as in Table 4. Because the word "mytelkomsel" is one of the word that is semantically similar to the word "telkomsel" and the word "mytelkomsel" is also found in tweet content, the value of the "telkomsel" feature in the tweet representation vector is replaced with the weight value of the word "mytelkomsel" as illustrated in Table 5. This applies when feature extraction uses TF-IDF, but if using BOW the value is replaced with 1[11].

Table 5. Vector Representation on Tweet Before and After Feature Expansion

Before	aplikasi	mytelkomsel	gangguan	telkomsel
	0.945	0.926	0.894	0
After	aplikasi	mytelkomsel	gangguan	telkomsel
	0.945	0.926	0.894	0.926

In table 5 it can be seen, after feature expansion, the value of the vector representation of the word "telkomsel" which was originally 0 was replaced by 0.926 because the word "telkomsel" has similarity with the word "mytelkomsel".

2.6 Support Vector Machine

Support Vector Machine (SVM) is one of the methods of supervised learning that analyzes data and recognizes patterns from data which is usually used for classification and regression[15]. SVM is effective method for text classification and has the advantage of being able to handle very large data in classification, especially text classification[16]. The way SVM works is find the best hyperplane by maximizing the margin that can generally separate the dataset into classes. The illustration of how SVM works is shown in Figure 2.

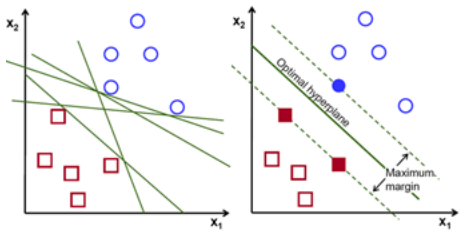

Figure 2. SVM Find The Best Hyperplane[17]

The basic principle of SVM is for linear classification, but nowadays the concept of kernel trick has been developed to solve non-linear problems. In this paper, four types of kernels are used, namely linear, Radial Basis Function (RBF), polynomial, and sigmoid. From the four kernels will be tested with a certain ratio and one with the best performance is taken. The kernel with the best ratio will be used for the next scenario experiment. Table 6 shows the SVM kernels and their formulas.

Table 6. SVM Kernels[18]

Kernel Type	Formula
Linear	$K(x_i, x_k) = {x_k}^T x$
RBF	$K(x_i, x_k) = \exp\{\frac{-\lvert\lvert x - x_k \rvert\rvert^2}{2\sigma^2}\}$
Polynomial	$K(x_i, x_k) = ({x_k}^T + 1)^d$
Sigmoid	$K(x_i, x_k) = \tanh[k{x_k}^T x + \theta]$

2.7 Performance Evaluation

To measure the quality of the system that has been created, a performance evaluation of the system is required. In this paper, we evaluate F1-score of the classification model that has been created using a confusion matrix. F1-Score is used because the data used in this paper is imbalance data. According to Ibrahim's research[19], performance measurement with F1-score is better for imbalance data than other measurements such as accuracy or ROC AUC. On imbalance data, accuracy can cause bias for some cases, and ROC AUC allows masking poor performance. Therefore, this paper uses F1-score for performance evaluation.

In this paper, True Positive (TP) is the result of the classification that is predicted positive and the result is true in reality, True Negative (TN) is the result that we predict negative and true in reality, False Positive (FP) is the result that we predict positive but wrong in reality, False Negative (FN) is the result that we predict negative but wrong in reality.

Before measuring the F1-score, we must first find the values for precision and recall. Precision is a measure of accuracy in proving the positive class in the actual value according to the positive class predicted previously. Formula 2 was used to calculate the precision.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

DOI: https://doi.org/10.29207/resti.v6i4.4187

Recall is a measure of the ability to correctly predict a positive class from all positive classes in an act value. Formula 3 was used to calculate the recall.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

After obtaining the precision and recall values, then we can find the F1-Score value. F1-Score is the harmonic mean between precision and recall. Formula 4 was used to calculate the F1-Score.

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (4)$$

3. Results and Discussions

The purpose of this research is to find the F1-Score of SVM classification with feature expansion using Fasttext for each aspect. There are 2 aspects taken in this experiment, the signal aspect and the service aspect. This experiment is divided into 4 scenarios, and each scenario tests each individual aspect. The first scenario, we compare F1-Score of SVM Classifier with random ratio 70:30, 80:20, and 90:10 on the training data and test data using TF-IDF and the default kernel of SVM used is RBF kernel. The best ratio will be used for F1-Score comparison between other SVM kernels. Then, the best ratio and kernel will be used as the baseline and will be used continuously for the next scenario. The second scenario is to conduct data sampling to overcome data imbalance using undersampling technique and SMOTE technique. Then, the third scenario measures the performance effect of using feature expansion using Fasttext with three corpuses that have been previously created. In the third scenario, each corpus is tested for Top 1, Top 5, Top 10, Top 20, and Top 25 features. The meaning of Top 1 feature is it will take a feature or a word from corpus that similar with a word in the tweet, and it applies to all features. Lastly, the fourth scenario performs hyperparameter tuning to improve model performance.

In each classification system for each scenario, the program is executed 5 times and the average F1-Score value is taken in order to obtain optimal F1-Score results.

3.1 Results

For the baseline search in scenario 1 is done through two stages. The first stage of scenario 1 is find the best data ratio. The results of the first stage of the first scenario are shown in Table 7. Table 7 shows the comparison of the F1-Score of SVM model using the default kernel from SKLearn's SVC library, the Radial Basis Function (RBF) kernel with three combinations random ratio of training data and test data of 70:30, 80:20, 90:10.

Table 7 shows the F1-score of each aspect of each ratio. For the signal aspect, the best ratio is 90:10 with an F1-score of 73.06%, and for the service aspect, the best F1-score is obtained at a ratio of 90:10 with a score of 64.46%. Then, the selected ratio will be used for F1-score comparison with three other SVM kernels.

Table 7. The Best Data Ratio for Baseline

Aspect	Data Ratio	F1-Score (%)
Signal	70:30	71.33
	80:20	72.46
	90:10	**73.06**
Service	70:30	63.60
	80:20	64.20
	90:10	**64.46**

Table 8. The Best Kernel for Baseline

Aspect	Data Ratio	Kernel	F1-Score (%)
Signal	90:10	Linear	**75.00**
		Polynomial	67.33
		RBF	73.06
		Sigmoid	74.00
Service		Linear	**66.40**
		Polynomial	48.46
		RBF	64.46
		Sigmoid	64.53

Table 8 shows the results of comparing the F1-score of the four SVM kernels for each aspect. For the signal aspect and the service aspect both get the best score on the linear kernel. So, the baseline is obtained on the linear kernel and data ratio 90:10. Then, it will be used for the next scenario test.

In the second scenario, because of the data used is imbalance, sampling technique will be carried out to overcome the data imbalance problem. The sampling stage will test two sampling technique, which are undersampling technique and SMOTE technique. The performance of the two sampling technique will be compared, and the technique with the best results will be used for the next scenario. Table 9 shows the comparison between F1-score of SVM classification using undersampling and SMOTE for each aspect.

Table 9. Performance of SVM with Data Sampling

Sampling	F1-Score (%)	
	Signal	Service
Undersampling	85.13 (+13.51)	78,80 (+18.67)
SMOTE	**95.86 (+27.82)**	**92.80 (+39.75)**

For undersampling technique, the signal aspect gets an increase in F1-score to 85.13% and the service aspect increased to 78.80%. Meanwhile, sampling with SMOTE technique showed an increase in F1-score as well, with the signal aspect increasing to 95.86% and the service aspect increasing to 92.80%. Since SMOTE shows the best performance, it will be used for testing in the next scenario.

Then, entering the third scenario, feature expansion was carried out using Fasttext with three different corpuses that had been previously created. Table 10 and Table 11 shows the F1-score results for each corpus based on its Top-n features.

DOI: https://doi.org/10.29207/resti.v6i4.4187

Table 10. Performance of SVM with Fasttext Feature Expansion for Signal Aspect

Features	F1-Score (%)		
	Tweets Corpus	IndoNews Corpus	Tweets+IndoNews Corpus
Top-1	95.33 (+27.11)	95.47 (+27.29)	95.73 (+27.64)
Top-5	94.93 (+26.58)	**95.92 (+27.90)**	95.67 (+27.56)
Top-10	94.93 (+26.58)	95.87 (+27.82)	95.87 (+27.82)
Top-20	94.60 (+26.13)	95.90 (+27.86)	95.80 (+27.73)
Top-25	94.57 (+26.10)	95.80 (+27.73)	95.79 (+27.71)

Table 11. Performance of SVM with Fasttext Feature Expansion for Service Aspect

Features	F1-Score (%)		
	Tweets Corpus	IndoNews Corpus	Tweets+IndoNews Corpus
Top-1	92.07 (+38.65)	92.27 (+38.96)	92.40 (+39.16)
Top-5	92.40 (+39.16)	92.87 (+39.86)	93.00 (+40.06)
Top-10	93.07 (+40.16)	93.47 (+40.76)	93.40 (+41.66)
Top-20	93.40 (+40.66)	93.73 (+41.16)	**93.80 (+41.27)**
Top-25	93,33 (+40.56)	93.64 (+41.03)	93.40 (+40.66)

In Table 10 and Table 11, it can be seen for the signal aspect after applying feature expansion, the best F1-Score is obtained by Top 5 in the IndoNews corpus with a value of 95.92%, it means increase of 27.90% from the baseline model. Meanwhile, for the service aspect, the best F1-Score is obtained by the Top 20 in the Tweets+Indonews corpus with a value of 93.80%, an increase of 41.27% from the baseline model. Figure 7 shows the performance comparison for each corpus on each Top-n feature.

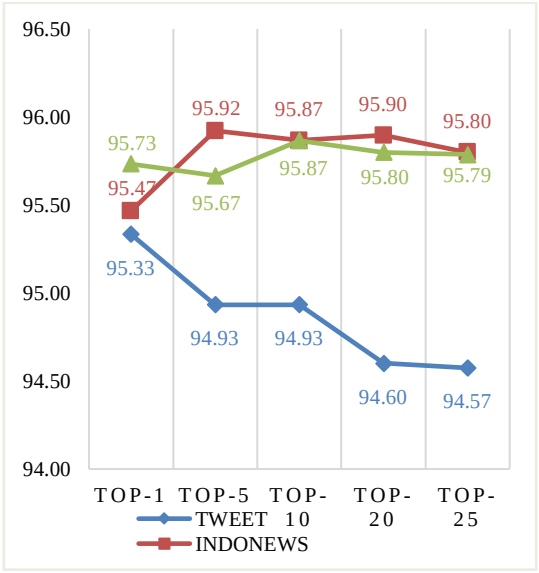

Figure 3. Comparison of F1-score After Feature Expansion for Signal Aspect

Furthermore, the last scenario will be carried out hyperparameter tuning. For hyperparameter tuning C and Gamma parameters will be used. The value of C and Gamma will be searched using the grid search technique, so that the best value of C and Gamma for parameters are obtained. Table 12 shows the combination of parameter values that are used in the hyperparameter tuning process.

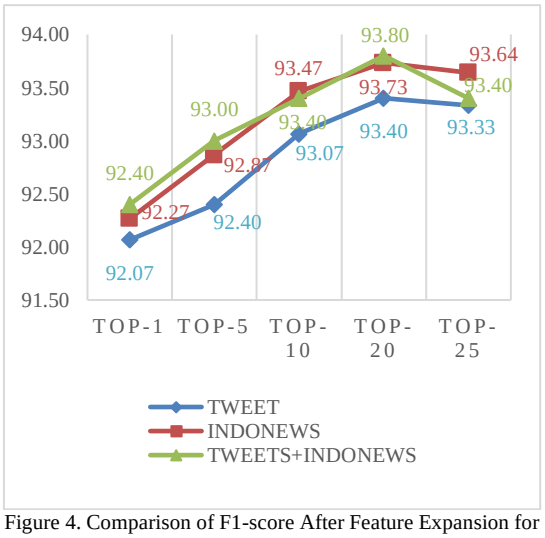

Figure 4. Comparison of F1-score After Feature Expansion for Service Aspect

Table 12. Parameter for Hyperparameter Tuning

Parameter	Values
C	[0.1, 0.5, 1, 5, 10]
Gamma	['scale','auto']

To help the hyperparameter tuning process, SKLearn's GridSearchCV library is used in the Python programming language which will provide output in the form of a combination of C and Gamma values that produce the best performance.

The best parameters for the signal aspect are obtained at C=1 and Gamma="scale", while for the service aspect, the optimal parameters are obtained at C=5 and Gamma="scale". Table 13 shows the performance of the model after hyperparameter tuning for each aspect.

Table 13. The Best Parameter of Hyperparameter Tuning

Aspect	C	Gamma	F1-Score (%)
Signal	1	Scale	95.93 (+27.91)
Service	5	Scale	94.53 (+42.36)

Table 13 shows the results of hyperparameter tuning with the previously mentioned parameters made the performance increase. For the signal aspect, the F1-Score value increased by 27.91% from the baseline model to 95.93% and for the service aspect the F1-Score value increased by 42.36% from the baseline model to 94.53%.

3.2. Discussions

Based on the results of the four scenarios that have been carried out as in Tables 7-11, it shows that implementing feature expansion using Fasttext word embedding can improve the performance of sentiment models. The performance improvement of the model is also influenced by the use of data sampling using SMOTE technique that used to overcome data imbalance and optimization with hyperparameter tuning using grid search to get more optimal classification parameter. Table 14 shows a comparison

DOI: https://doi.org/10.29207/resti.v6i4.4187

of the results of the four scenarios that have been carried out for each aspect.

Table 14. Comparison of Experiment Results

Aspect	Experiment	F1-Score (%)
Signal	Baseline	75.00
	Baseline + SMOTE	95.87 (+27.82)
	Baseline + SMOTE + Feature Expansion	95.92 (+27.90)
	Baseline + SMOTE + Feature Expansion + Hyperparameter Tuning	95.93 (+27.91)
Service	Baseline	66.40
	Baseline + SMOTE	92.80 (+39.75)
	Baseline + SMOTE + Feature Expansion	93.80 (+41.27)
	Baseline + SMOTE + Feature Expansion + Hyperparameter Tuning	94.53 (+42.36)

In Table 14, it can be seen that there is a significant increase in performance for both aspects in each scenario that has been carried out. For the signal aspect, the performance for the baseline is 75%, while for the service aspect it is 66.40%. Then, when implementing data sampling with SMOTE, the F1-Score value for each aspect experienced a fairly high increase of 95.87% for the signal aspect and 92.80% for the service aspect. In third scenario, which is the implementation of feature expansion, the F1-score of both aspects increased to 95.92% for the signal aspect and 93.80% for the service aspect. Lastly, in the fourth scenario, hyperparameter tuning was carried out and there was an increase in F1-score again with a score of 95.93% for the signal aspect and 94.53% for the service aspect.

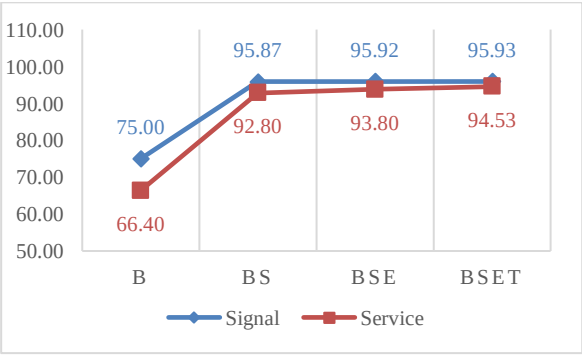

Figure 5. Performance Improvement Results on Each Scenario of Each Aspect

To visualize the results from Table 14, see Figure 5. In Figure 5, the blue line is the signal aspect and the red line is the service aspect. Initial B is for scenario 1, namely Baseline. Initial BS is for scenario 2, namely Baseline + SMOTE. Initial BSE is for scenario 3, namely Baseline + SMOTE + Feature Expansion. Initial BSET is for scenario 4, namely Baseline + SMOTE + Feature Expansion + Hyperparameter Tuning. From Figure 5, it can be seen that there is a significant increase in performance in each scenario for each aspect.

4. Conclusion

In this paper we have done research on implementation Fasttext word embedding as feature expansion and SVM classification in aspect-based sentiment analysis. This research takes a case study of Telkomsel's products, a cellular company in Indonesia for signal aspect and service aspect. The data used in the study was taken from twitter tweets by scrapping using the SNScrape library in the python programming language. The data successfully taken was 16.978 tweets, and also IndoNews data to help the feature expansion process. Based on the 4 scenarios that have been carried out, using sampling with SMOTE technique produces better performance than using undersampling technique. Then, the implementation of feature expansion in the model shows an increase in performance for each aspect. The use of hyperparameter tuning also helps to improve the model's performance results. The final result obtained for the signal aspect achieved the best F1-Score value on the IndoNews corpus in the Top-5 features of 95.93%, and the service aspect achieved the best F1-Score value on the Tweets+IndoNews corpus in the Top-20 of 94.53%. For future research, perhaps the research can be done using a larger amount of data and take more aspects than this research, and can also apply the other classification methods.

References

[1] J. E. Sembodo, E. B. Setiawan, and Z. K. Abdurahman Baizal, "The improvement of Indonesian news curator classification in Twitter," *2017 5th Int. Conf. Inf. Commun. Technol. ICoIC7 2017*, vol. 0, no. c, 2017, doi: 10.1109/ICoICT.2017.8074658.

[2] E. B. Setiawan, D. H. Widyantoro, and K. Surendro, "Feature Expansion using Word Embedding for Tweet Topic Classification," *2016 10th Int. Conf. Telecommun. Syst. Serv. Appl.*, no. 2011, 2016, doi: 10.1109/TSSA.2016.7871085.

[3] S. Thavareesan and S. Mahesan, "Sentiment Analysis in Tamil Texts: A Study on Machine Learning Techniques and Feature Representation," *2019 IEEE 14th Int. Conf. Ind. Inf. Syst. Eng. Innov. Ind. 4.0, ICIIS 2019 - Proc.*, pp. 320–325, 2019, doi: 10.1109/ICIIS47346.2019.9063341.

[4] P. Mehta and S. Pandya, "A review on sentiment analysis methodologies, practices and applications," *Int. J. Sci. Technol. Res.*, vol. 9, no. 2, pp. 601–609, 2020.

[5] R. Hajrizi and K. P. Nuçi, "Aspect-Based Sentiment Analysis in Education Domain," no. October, 2020, [Online]. Available: http://arxiv.org/abs/2010.01429.

[6] N. U. Pannala, C. P. Nawarathna, J. T. K. Jayakody, L. Rupasinghe, and K. Krishnadeva, "Supervised learning based approach to aspect based sentiment analysis," *Proc. - 2016 16th IEEE Int. Conf. Comput. Inf. Technol. CIT 2016, 2016 6th Int. Symp. Cloud Serv. Comput. IEEE SC2 2016 2016 Int. Symp. Secur. Priv. Soc. Netwo*, pp. 662–666, 2017, doi: 10.1109/CIT.2016.107.

[7] Irbah Salsabila and Yuliant Sibaroni, "Multi Aspect Sentiment of Beauty Product Reviews using SVM and Semantic Similarity," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 3, pp. 520–526, 2021, doi: 10.29207/resti.v5i3.3078.

[8] J. W. Iskandar and Y. Nataliani, "Perbandingan Naïve Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 6, pp. 1120–1126, 2021, doi: 10.29207/resti.v5i6.3588.

[9] I. Kaibi, E. H. Nfaoui, and H. Satori, "A comparative

DOI: https://doi.org/10.29207/resti.v6i4.4187

evaluation of word embeddings techniques for twitter sentiment analysis," *2019 Int. Conf. Wirel. Technol. Embed. Intell. Syst. WITS 2019*, pp. 1–4, 2019, doi: 10.1109/WITS.2019.8723864.

[10] B. Gupta, M. Negi, K. Vishwakarma, G. Rawat, and P. Badhani, "Study of Twitter Sentiment Analysis using Machine Learning Algorithms on Python," *Int. J. Comput. Appl.*, vol. 165, no. 9, 2017, doi: 10.5120/ijca2017914022.

[11] Alvi Rahmy Royyan and Erwin Budi Setiawan, "Feature Expansion Word2Vec for Sentiment Analysis of Public Policy in Twitter," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 1, pp. 78–84, 2022, doi: 10.29207/resti.v6i1.3525.

[12] N. A. N and E. B. Setiawan, "Implementation Word2Vec for Feature Expansion in Twitter Sentiment Analysis," no. 10, pp. 837–842, 2021.

[13] N. Nedjah, I. Santos, and L. de Macedo Mourelle, "Sentiment analysis using convolutional neural network via word embeddings," *Evol. Intell.*, pp. 2–6, 2019, doi: 10.1007/s12065-019-00227-4.

[14] A. Nurdin, B. Anggo Seno Aji, A. Bustamin, and Z. Abidin, "Perbandingan Kinerja Word Embedding Word2Vec, Glove, Dan Fasttext Pada Klasifikasi Teks," *J. Tekno Kompak*, vol. 14, no. 2, p. 74, 2020, doi: 10.33365/jtk.v14i2.732.

[15] S. K. Lidya, O. S. Sitompul, and S. Efendi, "Sentiment Analysis Pada Teks Bahasa Indonesia Menggunakan Support Vector Machine (Svm)," *Semin. Nas. Teknol. dan Komun. 2015*, vol. 2015, no. Sentika, pp. 1–8, 2015, doi: 10.1016/j.eswa.2013.08.047.

[16] Oryza Habibie Rahman, Gunawan Abdillah, and Agus Komarudin, "Klasifikasi Ujaran Kebencian pada Media Sosial Twitter Menggunakan Support Vector Machine," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 1, pp. 17–23, 2021, doi: 10.29207/resti.v5i1.2700.

[17] G. D. Salsabila and E. B. Setiawan, "Semantic Approach for Big Five Personality Prediction on Twitter," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 4, pp. 680–687, 2021, doi: 10.29207/resti.v5i4.3197.

[18] A. R. D. Pratiwi and E. B. Setiawan, "Implementation of Rumor Detection on Twitter Using the SVM Classification Method," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 5, pp. 782–789, 2021, doi: 10.29207/resti.v4i5.2031.

[19] M. Ibrahim, M. Torki, and N. El-Makky, "Imbalanced Toxic Comments Classification Using Data Augmentation and Deep Learning," *Proc. - 17th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2018*, pp. 875–878, 2019, doi: 10.1109/ICMLA.2018.00141.

DOI: https://doi.org/10.29207/resti.v6i4.4187