Journal of Student Research Exploration

https://shmpublisher.com/index.php/josre
p-ISSN 2964-8246| e-ISSN 2964-1691

Sentiment analysis of youtube comments on the palestine-israel conflict: Performance comparison of SVM, KNN, and RFC

Irendra Lintang Keksi[1], Jumanto[2], Amin Padmo Azam Masa[3]
[1, 2] Department of Computer Science, Universitas Negeri Semarang, Indonesia
[3]Information Systems Study Program, Universitas Mulawarman, Indonesia

Article Info

Article history:

Received July 13, 2024
Revised March 14, 2025
Accepted March 17, 2025

Keywords:

Conflict
Sentiment
Youtube
RFC
Evaluation

ABSTRACT

The Palestine-Israel conflict, rooted in territorial and religious identity disputes in the Middle East, notably over the sanctity of Jerusalem, is impacted by various political, economic, and social factors. This study employs text-mining techniques to analyze the sentiment of YouTube comments concerning the conflict. Utilizing data collected via the YouTube API, the study preprocesses, analyzes sentiment, and classifies comments using three machine learning algorithms: K-Nearest Neighbors (K-NN), Random Forest Classifier (RFC), and Support Vector Machine (SVM). The categorization report measures are utilized to compare how well the models performed in classifying estimation as positive or negative. Outflanking all other classifiers, the Irregular Woodland Classifier (RFC) accomplishes 78curacy with accuracy rates of 0.76 for positive and 0.79 for negative assumptions. With a precision rate of 77%, SVM illustrates an inclination in favor of negative sentiments, though K-NN, with an exactness rate of 60%, shows an imbalance favoring negative over positive estimations.

1. Introduction

[1] ***Corresponding Author:***
Irendra Lintang,
Department of Computer Science,
Universitas Negeri Semarang,
Sekaran, Gunung Pati, Kota Semarang, Indonesia.
Email: irendralintang@students.unnes.ac.id
DOI: https://doi.org/10.52465/josre.v3i1.426

One of the foremost perplexing and extended discussions in modern history is the one between Palestine and Israel, which is based on Center Eastern devout character and geological clashes. There are a few political, financial, and social perspectives to this war, such as the progressing attack of Gaza, Israeli settlement extension within the West Bank, Palestinian outcast status, and human rights infringement. Jerusalem's part as a sacrosanct city for Judaism, Islam, and Christianity raises the conflict's complexity and makes it a center point for mindfulness around the world.

Various things, such as that of Ibnu Afdhal et al. (2022), have been done on estimation assessment utilizing social media stages. Collecting information from YouTube comments on recordings that contain Islamophobia is considered an estimation examination [1]. Sites like YouTube have become a major stage for people of all foundations to voice their thoughts as social media utilization has expanded. Numerous societal perspectives and sentiments are reflected in comments cleared out on recordings of the Palestine-Israel struggle. Analyzing the opinions of these comments can uncover vital subtle elements of how the open recognizes the strife.

This work applies SVM, K-NN, and RFC classification approaches to sentiment evaluation. Maulidah (2023) analyzed sentiment regarding the Gojek and Grab super applications using Twitter data, a random forest classifier algorithm, a support vector machine, and long short-term memory techniques. The study discovered that the Support Vector Machine performed best, with an accuracy value of 0.92 and an F1-Score of 0.90 on the Gojek dataset [2].

Sentiment analysis is a computational technique for determining polarity (positive or negative). This study uses sentiment analysis on YouTube comments pertaining to the conflict between Israel and Palestine. The research uses text-mining techniques to examine the tone of YouTube comments about the conflict between Israel and Palestine.

This research is expected to identify the most accurate method for analyzing public opinion regarding the disputes between Palestine and Israel by understanding the sentiment in online communities.

The following are the hypotheses of the research:

- Public sentiment towards the Palestine-Israel conflict tends to be negative, as reflected in YouTube comments describing Israel's actions in the war against Palestine.
- The Random Forest algorithm performs better than the SVM and K-NN algorithms in classifying the sentiment of YouTube comments related to the Palestine-Israel conflict.

2. Method

This research utilizes text-mining techniques combined with sentiment analysis and machine learning classification. This process involves several key stages.

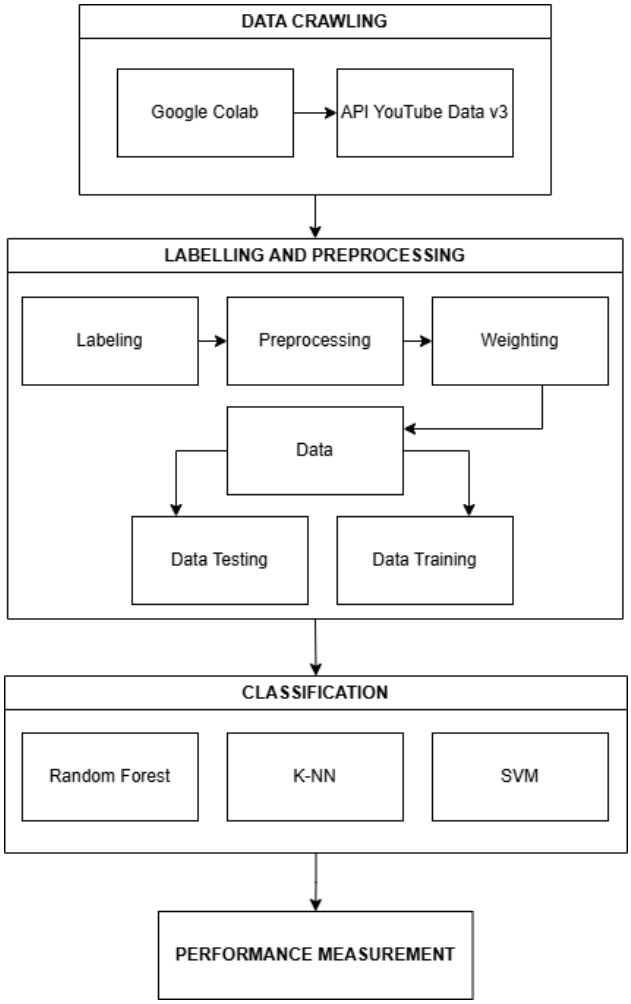

Figure 1. The stages of research in sentiment analysis

Text Mining

Text mining is an automated process used to extract useful information from unstructured text. Typically, the way text mining works adopts many principles from data mining, with the main difference lying in the data sources used. Text mining works with data from natural language processing (NLP) that is unstructured, while data mining refers to patterns extracted from structured databases [3]. The text mining process involves several key stages, including data collection, text cleaning, transforming data into a processable format, analyzing text using algorithms, and interpreting the results for decision-making.

Data Crawling

In this research, a data crawling technique is applied to collect comments from the YouTube video "Why Israel is in deep trouble: John Mearsheimer with Tom

Switzer" from the Centre for Independent Studies YouTube channel using the YouTube Data API v3. Within the crawling stage, analysts utilized Google Colab Python to get roughly 3,000 comments [4]. An API key was used to set up the YouTube API and confirm demands. Utilizing the panda's library, the collected information was saved in a DataFrame and sent out to a CSV record for extra examination.

Data Labeling

After being collected utilizing web scratching, the information will be labeled. In arrange to recognize between positive and negative reviews, each survey within the dataset is labeled for investigation [5]. In this step, patterns within the content are recognized and names are consequently assigned by the application of machine learning calculations and natural language processing (NLP) strategies.

Data Preprocessing

Preprocessing may be an arrangement of preliminary activities in text processing intended to change over reports into organized information when essential, empowering their consequent utilization in text mining [6]. Significant activities are performed during the data preprocessing stage to ensure the comment writings are arranged for estimation examination. By taking these steps, unessential components and conceivable inconvenient impacts on the analysis's discoveries are diminished.

Case folding

The procedure of changing each letter in a content to lowercase is known as case folding. This step is performed to make content assessment less demanding. Standardizing the letter cases requires case folding since not all recovered comment information is reliably in lowercase [7].

Tokenization

Tokenization is utilized to separate sentences into their component words [8]. Tokens are the little parts delivered by the content division. After that, each token is spared in a suitable organization for further examination.

Filtering

The method of dispensing with unneeded or insignificant literary parts is known as filtering. Accentuation, digits, and other extraordinary characters that do not include much to text examination will be disposed of.

Stopwords removal

To move forward the analysis's precision, all words that can be regarded pointless are dispensed with at this point [9]. Connector words that are regularly utilized in

writings but do not truly offer assistance the reader understand the substance are known as stopwords.

Stemming

The method of stemming includes taking words down to their most essential frame and eliminating joins based on the reason of the word [10]. After filtering, this step moreover decides each word's root [11].

Weighting

The weighting method gives each word in a report or dataset weight esteem, agreeing to its noteworthiness. Words with visit utilization but small data are assigned lower weights. In contrast, exceptional words with critical implications are assigned bigger weights. By calculating a word's recurrence of events in a document (TF) and considering its notoriety over all archives (IDF), weighting utilizing TF-IDF is decided. In a given document, a word's TF score increases with its frequency of occurrence. In contrast, its IDF score increases with its rarity [12]

$$W_{t,d} = (1 + {}^{10}\log tf)\left(\frac{{}^{10}\log^{n}}{df_t}\right) \quad (1)$$

Explanation:

$W_{t,d}$ = Final weight value of the word.
tf = Number of occurrences of the word in the document.
n = Total number of documents.
df_t = Number of documents containing the word.

Data and Sample

Sentiment analysis has been used to the collected and processed comment data from YouTube videos. Three thousand comments make up the dataset, which has been preprocessed to purify and standardize the text. The dataset is divided into training and testing sets following the preparation phase. The train_test_split function from the Scikit-Learn library is used to carry out this separation procedure [13].

90% of the dataset is utilized for training purposes. In particular, 2,700 of the 3,000 data points are used for training, and the remaining 300 data points, or 10% of the entire dataset gathered, are set aside for testing.

Data Attributes

Three primary attributes comprise the data used in this study: text, sentiment, and sentiment_class. With each row representing an unanalyzed comment, the text attribute holds original comments that were taken from YouTube videos. The sentiment polarity score is then calculated for every comment in the text column, with the values resulting in the sentiment column. Sentiment polarity values shift from -1.0 to 1.0; positive values indicate positive opinion, whereas negative and

zero values indicate negative opinion. Utilizing this polarity scale, each comment is classified into one of two estimation classes ('Negative' for polarity values equal to or underneath 0 and 'Positive' for polarity values over 0), which are kept within the sentiment_class column.

Classification

Within the spaces of statistics and machine learning, classification could be a strategy for organizing information into preset classes [14]. Using patterns or characteristics found from previously labeled information, it looks for to make a calculation that will be utilized to anticipate categories of obscure data.

K-nearest neighbors (K-NN)

Information are categorized utilizing the K-Nearest Neighbors (K-NN) strategy, which does so by comparing each information point to another [15]. As a rule, the Euclidean distance is utilized to decide this proximity.

Random forest classifier (RFC)

Decision trees are utilized as classifiers within the combination approach known as Random Forest. A decision tree inside a random forest produces a majority class, which decides the ultimate course designated within the classification step [16]. Each decision tree is built from an arbitrary subset of the training data using the bootstrap inspecting procedure. An irregular subset of highlights is picked at each tree node, and the node is partly based on the finest highlight [17]. The ideal feature is chosen based on specific parameters, like information gain.

Support vector machine (SVM)

To decide the ideal separating line between information classes, the classification strategy Support Vector Machine (SVM) is utilized. The hyperplane could be a line that has been chosen to maximize the edge, which is the distance between the closest information focuses from each class and this separating line. Support vectors allude to the data points that are closest to the hyperplane [18].

Evaluation

Model evaluation may be a prepare that looks at a model's adequacy by utilizing information created by the model to see how well it can classify recently found information. When analyzing the effectiveness of a classification model within the setting of machine learning classification issues, a confusion matrix is an evaluation matrix that might abdicate two or more classes as results [19]. Its essential objective is to compare test data results from the classification system with the genuine classification targets. Model evaluation is conducted employing a confusion matrix that measures different perspectives of demonstrate execution, such as precision, accuracy, recall, and F1-score. For occasion, accuracy in measuring the execution of a classification show utilizing the confusion matrix

strategy is calculated by looking at the number of accurately classified information focuses against the whole number of information focuses [20].

$$Accuracy = \frac{TP + TN}{TP+TN+FP+FN} \times 100\% \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

Explanation:

TP = True Positive value

TN = True Negative value

FP = False Positive value

FN = False Negative value

3. Results and Discussion

This section reviews the results and analysis of the performance of each algorithm using various classification methods applied in this study. After completing the data crawling process, 3,000 comment data were obtained and the sentiment polarity scores were calculated. Subsequently, labeling was performed with positive and negative labels based on these polarity scores.

Crawling Data

Comments were retrieved based on chronological order, with the most recent ones obtained first. These comments were then stored in a dataframe format using the pandas library and exported to a CSV file.

Table 1. Sample of crawled data results

	Text
1	Towards the end of the session where the...
2	Wonderful talk. Clear, concise, providing...
3	very interesting. thanks for sharing
4	A RARE American who has the guts and...
...	...
3000	Israel is facing a grim denouement. The...

Labeling

Data labeling was performed using the TextBlob library. The sentiment polarity of each comment was then calculated. This polarity value was applied to the comment text column. The sentiment of each comment was classified into two

categories: if the polarity value is above 0, it is labeled as "Positive", and if the polarity value is less than or equal to 0, the sentiment is classified as "Negative".

Table 2. Labeled data samples

Text	Sentiment
Towards the end of the session where the...	Positive
Wonderful talk. Clear, concise, providing...	Positive
very interesting. thanks for sharing	Positive
A RARE American who has the guts and...	Positive
...	...
Israel is facing a grim denouement. The...	Negative

Data Preprocessing

The initial data processing involves a series of steps, namely case folding, tokenization, filtering, stopwords removal, and stemming. Below is an example of the dataset after each preprocessing step has been applied:

Table 3. Sample of tokenization results

Tokenized Text
['towards', 'the', 'end', 'of', 'the', 'session', 'where', 'the',...]
['wonderful', 'talk', 'clear', 'concise', 'providing',...]
['very', 'interesting', 'thanks', 'for', 'sharing']
['a', 'rare', 'american', 'who', 'has', 'the','guts','and',...]
...
['israel', 'is', 'facing', 'a', 'grim', 'denouement', 'the', 'middle',...]

Table 4. Preprocessed data sample

Preprocessed Text
toward end session student ask good professor...
wonder talk clear concis provid justif...
interest thank share
rare american gut decenc speak rude truth...
...
israel face grim denouement middle east final...

Weighting

The TF-IDF method assigns weights to each word in the document based on its occurrence within the document (TF) and its general occurrence across the entire document collection (IDF). By employing TF-IDF, text is represented as a feature vector that can be understood by machine learning algorithms such as Support Vector Machine (SVM), enabling text classification. The algorithms used are:

```
rf_pipeline = Pipeline([
 ('vectorizer', TfidfVectorizer()),
```

Evaluation

In this evaluation stage, we compile the model evaluation report using a confusion matrix to assess the performance of the three classification methods as follows:

K-nearest neighbor (K-NN)

Here are the performance evaluation results of the K-Nearest Neighbor classification method:

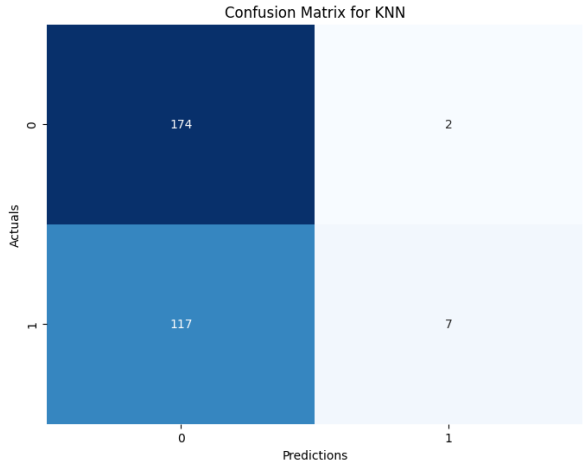

Figure 2. Confusion Matrix for KNN

Table 5. Confusion matrix results for the K-NN method

Confusion Matrix		Predicted Class	
Observed Class		**Positive**	**Negative**
	Positive	TP = 7	FN = 117
	Negative	FP = 2	TN = 174

Based on the confusion matrix, the K-NN method shows that the model has high accuracy in identifying negative cases, but performs less well in identifying positive cases.

Table 6. Evaluation of confusion matrix for K-NN method

	Positive	Negative
Recall	0.06	0.99
Precision	0.78	0.60
F1-Score	0.11	0.75
Accuracy	**60%**	

The K-NN classification method shows that the model tends to perform very well in recognizing negative sentiments with a precision of 0.60, recall of 0.99, and f1-score of 0.75. However, it performs poorly in recognizing positive sentiments with

a precision of 0.78, recall of 0.06, and f1-score of 0.11. The overall accuracy of the model is 0.60, indicating that the model more frequently classifies data as negative, resulting in an imbalance in detecting positive and negative sentiments.

Random forest classifier (RFC)

Here are the evaluation results of the Random Forest Classifier (RFC) classification method:

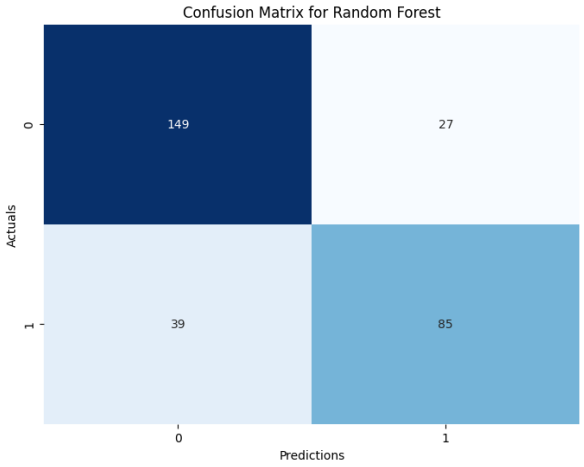

Figure 3. Confusion Matrix for RFC

Table 7. Confusion matrix results for the RFC method

Confusion Matrix		Predicted Class	
Observed Class		**Positive**	**Negative**
	Positive	TP = 85	FN = 39
	Negative	FP = 27	TN = 149

From the confusion matrix, the Random Forest Classifier (RFC) method shows that the model correctly identifies 85 positive cases (True Positive) and 149 negative cases (True Negative). All things considered, there are 39 positive cases that are erroneously classified as negative (False Negative) and 27 negative cases that are mistakenly classified as positive (False Positive). In recognizing both positive and negative cases, the model shows a great adjust.

Table 8. Confusion matrix evaluation for the RFC method

	Positive	Negative
Recall	0.69	0.85
Precision	0.76	0.79
F1-Score	0.72	0.82
Accuracy	**78%**	

The results of the sentiment examination method of the Random Forest Classifier (RFC) illustrate adjusted execution in distinguishing both positive and negative demeanors. This model is way better at distinguishing negative attitudes, with recall values of 0.85 for negative sentiment and 0.69 for positive emotion. The

precision values for positive and negative sentiment are 0.76 and 0.79, respectively, proposing a satisfactory degree of accuracy for both emotion categories. The model is more precise in classifying negative demeanors, as proven by the F1-Score, which combines precision and recall. It is 0.72 for positive sentiment and 0.82 for negative sentiment. The model performs well in sentiment categorization with a general accuracy of 78%.

Support vector machine (SVM)

The following are the Support Vector Machine (SVM) method evaluation results:

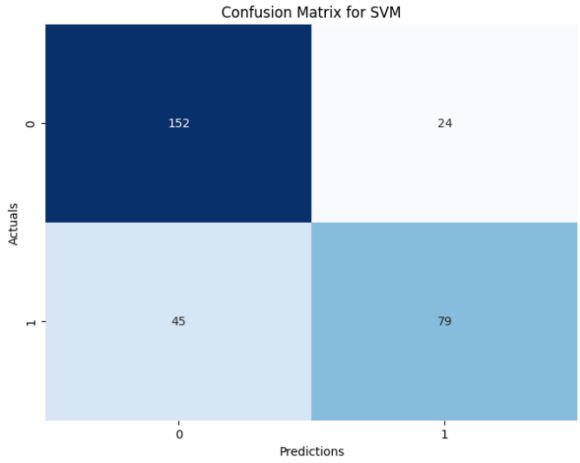

Figure 4. Confusion Matrix for SVM

Table 9. Results of the confusion matrix for the SVM method

Confusion Matrix		Predicted Class	
Observed Class		**Positive**	**Negative**
	Positive	TP = 79	FN = 45
	Negative	FP = 24	TN = 152

Agreeing with the confusion matrix collected from the SVM classification results, the model precisely decided 152 negative and 79 positive cases, but it erroneously categorized 24 negative cases as positive and 45 positive cases as negative. This recommends that even though the model recognizes the larger part of classes, it might still utilize a little work, especially in identifying less common groupings.

Table 10. Confusion matrix evaluation for the SVM method

	Positive	Negative
Recall	0.64	0.86
Precision	0.77	0.77
F1-Score	0.70	0.82
Accuracy	**77%**	

The evaluation results of sentiment using the Support Vector Machine (SVM) classification method depict a fairly good performance, with an overall accuracy of 77%. This model exhibits equal precision for both sentiments, scoring 0.77 for both

positive and negative sentiments, indicating a balanced level of accuracy in classifying both sentiment categories. However, the recall for negative sentiment (0.86) is slightly higher than for positive sentiment (0.64), suggesting that the model tends to be better at recognizing negative sentiments than positive sentiments. This is also reflected in the F1-score values, where negative sentiment has a slightly higher score (0.82) compared to positive sentiment (0.70).

Sentiment

Here are the numbers of positive and negative sentiments from the public regarding the YouTube video "Why Israel is in deep trouble: John Mearsheimer with Tom Switzer" from the Centre for Independent Studies YouTube channel:

Table 11. Number of sentiments

	Positive	Negative
Total	1.309	1.691
Percentage	43.63%	56.37%

Public sentiment tends to be more negative than positive. Out of the total number of analyzed comments, 56.37% of them were identified as negative, while only 43.63% were considered positive. This indicates that the video may have triggered many negative responses from viewers, possibly due to differences in opinion or disagreement with the issues discussed in the video.

Wordcloud Visualization

Wordcloud visualization is a graphical representation of text that displays frequently occurring words, using larger sizes to indicate their weight in the overall text [21]. Wordcloud visualizations for each sentiment from YouTube comment data are shown in the following image:

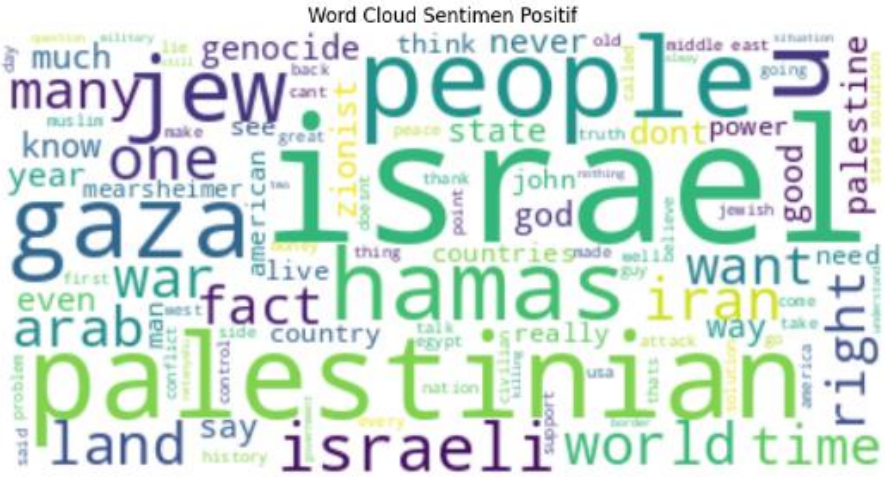

Figure 5. Visualization of positive sentiment wordcloud

The positive sentiment wordcloud also displays dominant words such as "Israel," "Palestinian," "Hamas," "Gaza," "people," and "world," albeit with a more

constructive and neutral connotation. Words like "right," "good," "peace," and "time" indicate that these comments are more focused on human rights, peaceful solutions, and understanding of the complex situation.

Figure 6. Visualization of negative sentiment wordcloud

The negative sentiment wordcloud shows domination of words such as "Israel," "Palestinian," "Hamas," "Iran," "Gaza," "war," "lie," and "genocide." This reflects that these comments focus on aspects of conflict, violence, and controversy in the relationship between Israel and Palestine. There is a lot of expression of dissatisfaction, anger, and criticism towards Israel's actions or policies, as well as tensions involving groups like Hamas and countries like Iran. In this case, the terms "terrorist" and "evil" suggest a distinct narrative of crime and violence.

4. Conclusion

With 56.37% of comments expressing negative emotion and 43.63% expressing positive sentiment, the sentiment analysis results of the YouTube video "Why Israel is in deep trouble: John Mearsheimer with Tom Switzer" reveal that public opinion is mostly negative toward the Israel-Palestine issue. Words that emphasize conflict and violence, like "war," "lie," and "genocide," are highlighted in the wordcloud representing negative attitude. On the other hand, the wordcloud representing positive sentiment shows terms like "good" and "peace," which represent conversations about solutions and human rights.

The Random Forest Classifier (RFC) outperformed K-NN and SVM in terms of algorithm performance, exhibiting 78% accuracy, precision of 0.76 for positive sentiment and 0.79 for negative sentiment, and a better balanced recall. The SVM algorithm also performed well with an accuracy of 77%, but showed a slight bias towards negative sentiment. On the other hand, K-NN demonstrated unbalanced performance with an accuracy of 60%, being very effective in detecting negative sentiment but poor in detecting positive sentiment. Overall, RFC proved to be the

best algorithm for sentiment classification in this context, supporting the hypothesis that RFC is more effective than SVM and K-NN in sentiment analysis of YouTube comments related to the Israel-Palestine conflict.

ACKNOWLEDGEMENTS

The author expresses gratitude to colleagues at Universitas Negeri Semarang who have consistently provided understanding and support in completing this research.

REFERENCES

[1] I. Afdhal, R. Kurniawan, I. Iskandar, R. Salambue, E. Budianita, dan F. Syafria, "Penerapan Algoritma Random Forest Untuk Analisis Sentimen Komentar Di YouTube Tentang Islamofobia," *J. Nas. Komputasi dan Teknol. Inf.*, vol. 5, no. 1, 2022, doi: https://doi.org/10.32672/jnkti.v5i1.4004.

[2] S. S. Maulidah, "Analisis Sentimen Terhadap Brand Reputation Super Apps Gojek dan Grab di Indonesia Menggunakan Algoritma Machine Learning," UIN Syarif Hidayatullah Jakarta, 2023.

[3] T. Ridwansyah, "Implementasi Text Mining Terhadap Analisis Sentimen Masyarakat Dunia Di Twitter Terhadap Kota Medan Menggunakan K-Fold Cross Validation Dan Naïve Bayes Classifier," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 2, no. 5, hal. 178–185, Apr 2022, doi: 10.30865/klik.v2i5.362.

[4] W. Widia, Z. Y. Aqsalia, S. Sari, N. U. Khoirunisa, dan F. Kurniawan, "Optimasi Algoritma Naive Bayes Untuk Menganalisis Sentimen Pada Konten Pemindahan Ibu Kota di Youtube," *J. Comput. Inf. Syst. Ampera*, vol. 5, no. 2, 2024, doi: https://doi.org/10.51519/journalcisa.v5i1.451.

[5] R. A. Saputra, D. P. Ray, dan F. Irwiensyah, "Analisis Sentimen Aplikasi Tokocrypto Berdasarkan Ulasan Pada Google Play Store Menggunakan Metode Naïve Bayes," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 4, 2024, doi: https://doi.org/10.30865/klik.v4i4.1707.

[6] T. Suhendra, B. Intan, dan A. T. Martadinata, "Analisis Sentimen Pengguna Aplikasi Netflix Pada Ulasan Google Playstore Menggunakan Metode Naïve Bayes," 2024.

[7] R. A. A. Renal, Syariful Alam, dan Moch Hafid T, "KOMPARASI PAYMENT DIGITAL UNTUK ANALISIS SENTIMEN BERDASARKAN ULASAN DI GOOGLE PLAYSTORE MENGGUNAKAN METODE SUPPORT VECTOR MACHINE," *STORAGE J. Ilm. Tek. dan Ilmu Komput.*, vol. 2, no. 3, hal. 118–128, Agu 2023, doi: 10.55123/storage.v2i3.2337.

[8] A. Nursalim dan R. Novita, "Sentiment Analysis of Comments On Google Play Store, Twitter And Youtube to The Mypertamina Application With Support Vector Machine," *J. Tek. Inform.*, vol. 4, no. 6, 2023, doi: https://doi.org/10.52436/1.jutif.2023.4.6.1059.

[9] S. Supriyatna dan E. Fahrudin, "Pemanfaatan Algoritma Text Mining dalam Menemukan Pola Risiko Bencana sebagai Pengetahuan Kebencanaan dari Dokumen Kajian Risiko Bencana (KRB)," *Jitu J. Inform. Utama*, vol. 2, no. 1, 2024, doi: https://doi.org/10.55903/jitu.v2i1.164.

[10] D. Wardhani, R. Astuti, dan D. D. Saputra, "Optimasi Feature Selection Text Mining: Stemming dan Stopword untuk Sentimen Analisis Aplikasi SatuSehat," *Innov. J. Soc. Sci. Res.*, vol. 4, no. 1, 2024, doi: https://doi.org/10.31004/innovative.v4i1.8759.

[11] Y. A. Putri Gabriella, "Optimasi Penerimaan Siswa Baru Dengan Penerapan Algortima Text Mining Dan Tf-Idf," *J. Comput. Informatics Res.*, vol. 2, no. 3, hal. 110–117, Jul 2023, doi: 10.47065/comforch.v2i3.941.

[12] N. Silalahi dan Guidio Leonarde Ginting, "Rekomendasi Berita Berkaitan dengan Menerapkan Algoritma Text Mining dan TF-IDF," *Bull. Comput. Sci. Res.*, vol. 3, no. 4, hal. 276–282, Jun 2023, doi: 10.47065/bulletincsr.v3i4.266.

[13] Jan Melvin Ayu Soraya Dachi dan Pardomuan Sitompul, "Analisis Perbandingan Algoritma

XGBoost dan Algoritma Random Forest Ensemble Learning pada Klasifikasi Keputusan Kredit," *J. Ris. RUMPUN Mat. DAN ILMU Pengetah. ALAM*, vol. 2, no. 2, hal. 87–103, Jul 2023, doi: 10.55606/jurrimipa.v2i2.1470.

[14] A. D. W. M. Sidik, I. Himawan Kusumah, A. Suryana, Edwinanto, M. Artiyasa, dan A. Pradiftha Junfithrana, "Gambaran Umum Metode Klasifikasi Data Mining," *Fidel. J. Tek. Elektro*, vol. 2, no. 2, hal. 34–38, Mei 2020, doi: 10.52005/fidelity.v2i2.111.

[15] L. Legito *et al.*, "Penerapan Algoritma K-Nearest Neighbor untuk Analisis Sentimen Terhadap Isu Khilafah dan Radikalisme di Indonesia," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 2, hal. 324–330, Nov 2023, doi: 10.57152/malcom.v3i2.893.

[16] A. P. Yudha dan R. P. Cahyono, "Analisis Kepuasan Pengunjung Menggunakan Metode Random Forest Untuk Wisata Pantai pada Pesawaran," *J. Ilmu Data*, vol. 2, no. 12, 2022.

[17] M. D. Purbolaksono, M. Irvan Tantowi, A. Imam Hidayat, dan A. Adiwijaya, "Perbandingan Support Vector Machine dan Modified Balanced Random Forest dalam Deteksi Pasien Penyakit Diabetes," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 2, hal. 393–399, Apr 2021, doi: 10.29207/resti.v5i2.3008.

[18] H. N. Irmanda dan Ria Astriratma, "Klasifikasi Jenis Pantun Dengan Metode Support Vector Machines (SVM)," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 5, hal. 915–922, Okt 2020, doi: 10.29207/resti.v4i5.2313.

[19] A. P. Giovani, A. Ardiansyah, T. Haryanti, L. Kurniawati, dan W. Gata, "ANALISIS SENTIMEN APLIKASI RUANG GURU DI TWITTER MENGGUNAKAN ALGORITMA KLASIFIKASI," *J. Teknoinfo*, vol. 14, no. 2, hal. 115, Jul 2020, doi: 10.33365/jti.v14i2.679.

[20] E. Fitriani, "PERBANDINGAN ALGORITMA C4.5 DAN NAÏVE BAYES UNTUK MENENTUKAN KELAYAKAN PENERIMA BANTUAN PROGRAM KELUARGA HARAPAN," *SISTEMASI*, vol. 9, no. 1, hal. 103, Jan 2020, doi: 10.32520/stmsi.v9i1.596.

[21] D. Wijaya, R. A. Saputra, dan F. Irwiensyah, "Analisis Sentimen Ulasan Aplikasi Samsat Digital Nasional Pada Google Playstore Menggunakan Algoritma Naïve Bayes," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 4, 2024.