DAFTAR PUSTAKA

Alhaq, Z., Mustopa, A., Mulyatun, S., & Santoso, J. D. (2021). Penerapan Metode Support Vector Machine Untuk Analisis Sentimen Pengguna Twitter. *Journal of Information System Management (JOISM)*, *3*(2), 44–49. https://doi.org/10.24076/joism.2021v3i2.558

Ali, Z., Hayat, M. F., Shaukat, K., Alam, T. M., Hameed, I. A., Luo, S., Basheer, S., Ayadi, M., & Ksibi, A. (2022). A Proposed Framework for Early Prediction of Schistosomiasis. *Diagnostics*, *12*(12), 1–25. https://doi.org/10.3390/diagnostics12123138

Alzamzami, F., Hoda, M., & Saddik, A. El. (2020). Light *Gradient Boosting* Machine for General Sentiment Classification on Short Texts: A Comparative Evaluation. *IEEE Access*, *8*, 101840–101858. https://doi.org/10.1109/ACCESS.2020.2997330

Ansar, A. (2020). The Unfolding of Belonging, Exclusion and Exile: A Reflection on the History of Rohingya Refugee Crisis in Southeast Asia. *Journal of Muslim Minority Affairs*, *40*(3), 441–456. https://doi.org/10.1080/13602004.2020.1819126

Ardianto, R., Rivanie, T., Alkhalifi, Y., Nugraha, F. S., & Gata, W. (2020). SENTIMENT ANALYSIS ON E-SPORTS FOR EDUCATION CURRICULUM USING NAIVE BAYES AND SUPPORT VECTOR MACHINE. *Jurnal Ilmu Komputer Dan Informasi (Journal of Computer Science and Information)*, *13*(2), 109–122. https://doi.org/10.1109/ICECOS47637.2019.8984530

Bahaaaldeen Abdul wahhab, A., & KareemabdulHassan, A. (2019). Proposed Aspect Based Sentiment Analysis system for English reviews. *Journal of Al-Qadisiyah for Computer Science and Mathematics*, *11*(2), 22–36. https://doi.org/10.29304/jqcm.2019.11.2.559

Chang, Y. S., Park, H. S., & Moon, I. J. (2021). Predicting the cochlear dead regions using a machine learning-based approach with *oversampling* techniques. *Medicina (Lithuania)*, *57*(11), 1–9. https://doi.org/10.3390/medicina57111192

Chen, M., Ubul, K., Xu, X., Aysa, A., & Muhammat, M. (2022). Connecting Text Classification with Image Classification: A New Preprocessing Method for Implicit Sentiment Text Classification. *Sensors*, *22*(5), 1899. https://doi.org/10.3390/s22051899

Chen, Y., & Yang, Z. (2021). Generalized eigenvalue proximal support vector machine for functional data classification. *Symmetry*, *13*(5), 1–16. https://doi.org/10.3390/sym13050833

D'Silva, R. I. (2021). Rohingya Refugees in South Asia: An Exploration of Social Borders and the Margins. *Borders in Globalization Review*, *3*(1), 35–45. https://doi.org/10.18357/bigr31202120261

Dikiyanti, T. D., Rukmi, A. M., & Irawan, M. I. (2021). Sentiment analysis and topic modeling of BPJS Kesehatan based on twitter crawling data using Indonesian Sentiment Lexicon and Latent Dirichlet Allocation algorithm. *Journal of Physics: Conference Series*, *1821*(1). https://doi.org/10.1088/1742-6596/1821/1/012054

Fajri, H. R. Al, Sinaga, R. B., Mubarok, H., Pangestu, A. D., & Prasvita, D. P. (2021).

Analisis Sentimen Pengguna Twitter terhadap Konflik antara Palestina dan Israel Menggunakan Metode Naïve Bayesian Classification dan Support Vector Machine. *Seminar Nasional Mahasiswa Ilmu Komputer Dan Aplikasinya (SENAMIKA)*, *September*, 166–175.

Ha, J., Kambe, M., & Pe, J. (2011). Data Mining: Concepts and Techniques. In *Data Mining: Concepts and Techniques*. https://doi.org/10.1016/C2009-0-61819-5

Hagmann, M., & Riezler, S. (2021). *False perfection in machine prediction: Detecting and assessing circularity problems in machine learning*. 1–31. http://arxiv.org/abs/2106.12417

Hamidi, G. D., Bestari, F. A., Situmorang, A., & Rakhmawati, N. A. (2021). Sentiment Analysis on the Ratification of Penghapusan Kekerasan Seksual Bill on Twitter. *Jurnal Teknik Informatika Dan Sistem Informasi*, *7*(3), 655–665. https://doi.org/10.28932/jutisi.v7i3.4051

Hashfi, F., Sugiarto, D., & Mardianto, I. (2022). Sentiment Analysis of An Internet Provider Company Based on Twitter Using Support Vector Machine and Naïve Bayes Method. *Ultimatics : Jurnal Teknik Informatika*, *4*(April), 1–6. https://doi.org/10.31937/ti.v14i1.2384

Idris, I. S. K., Mustofa, Y. A., & Salihi, I. A. (2023). Analisis Sentimen Terhadap Penggunaan Aplikasi Shopee Mengunakan Algoritma Support Vector Machine (SVM). *Jambura Journal of Electrical and Electronics Engineering*, *5*(1), 32–35. https://doi.org/10.37905/jjeee.v5i1.16830

Indrayuni, E., & Nurhadi, A. (2022). Sentiment Analysis About COVID-19 Booster Vaccine on Twitter Using Deep Learning. *SinkrOn*, *7*(3), 900–905. https://doi.org/10.33395/sinkron.v7i3.11485

Irawan, F. R., Jazuli, A., & Khotimah, T. (2022). Analisis Sentimen Terhadap Pengguna Gojek Menggunakan Metode K-Nearset Neighbors Sentiment Analysis of Gojek Users Using K-Nearest Neighbor. *JIKO (Jurnal Informatika Dan Komputer)*, *5*(1), 62–68. https://doi.org/10.33387/jiko

Isaeva, E., & Aldarova, Di. (2021). Text-mining in Terms of Methodology and Development. *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, 413–416. https://doi.org/10.1109/ElConRus51938.2021.9396437

Khaled, A. F. M. (2021). Do No Harm in refugee humanitarian aid: the case of the Rohingya humanitarian response. *Journal of International Humanitarian Action*, *6*(7), 1–13. https://doi.org/10.1186/s41018-021-00093-9

Liu, H., Chen, X., & Liu, X. (2022). A Study of the Application of Weight Distributing Method Combining Sentiment Dictionary and TF-IDF for Text Sentiment Analysis. *IEEE Access*, *10*, 32280–32289. https://doi.org/10.1109/ACCESS.2022.3160172

Montesinos López, O. A., Montesinos López, A., & Crossa, J. (2022). Multivariate Statistical Machine Learning Methods for Genomic Prediction. In *Multivariate Statistical Machine Learning Methods for Genomic Prediction*. https://doi.org/10.1007/978-3-030-89010-0

Muriyatmoko, D., Taufiqurrahman, T., & Humam, A. (2022). Analisis Sentimen

Masyarakat Terhadap Konflik Rusia dan Ukraina Menggunakan Metode Naïve Bayes pada Media Sosial Twitter. *Metik Jurnal*, *6*(2), 140–145. https://doi.org/10.47002/metik.v6i2.375

Nur-a-alam, Ahsan, M., Based, M. A., Haider, J., & Kowalski, M. (2021). COVID-19 detection from chest X-ray images using feature fusion and deep learning. *Sensors*, *21*(4), 1–30. https://doi.org/10.3390/s21041480

Purba, M., Ermatita, E., Abdiansah, A., Noprisson, H., Ayumi, V., Salamah, U., Setiawan, H., & Yadi, Y. (2022). Effect of Random Splitting and Cross Validation for Indonesian Opinion Mining using Machine Learning Approach. *International Journal of Advanced Computer Science and Applications*, *13*(9), 145–151. https://doi.org/10.14569/IJACSA.2022.0130917

Rachmatullah, M. I. C. (2022). The Application of Repeated SMOTE for Multi Class Classification on Imbalanced Data. *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, *22*(1), 13–24. https://doi.org/10.30812/matrik.v22i1.1803

Rahman, M. R., Faiz, M. A., Nu, M. Y., Hassan, M. R., Chakrabarty, A. K., Kabir, I., Islam, K., Jafarullah, A. K. M., Alakabawy, M., Khatami, A., & Rashid, H. (2020). A rapid assessment of health literacy and health status of rohingya refugees living in cox's bazar, Bangladesh following the august 2017 exodus from myanmar: A cross-sectional study. *Tropical Medicine and Infectious Disease*, *5*(3), 110. https://doi.org/10.3390/tropicalmed5030110

Rismawan, S. A., & Syahidin, Y. (2023). Implementasi Website Berita Online Menggunakan Metode Crawling Data Dengan Bahasa Pemrograman Python. *Jurnal Teknik Informatika Dan Sistem Informasi*, *10*(3), 167–178. https://doi.org/10.35957/jatisi.v10i3.4902

Ruslan, Zalmatin, W. O., & Syukur, S. (2023). Konflik Rohingya dan Pengakuan Kewarganegaraannya. *Edu Sociata: Jurnal Pendidikan Sosiologi*, *6*(2), 558–568.

Samuel, J., Ali, G. G. M. N., Rahman, M. M., Esawi, E., & Samuel, Y. (2020). COVID-19 public sentiment insights and machine learning for tweets classification. *Information (Switzerland)*, *11*(6), 314. https://doi.org/10.3390/info11060314

Sanjaya, M. O., Bukhori, S., & Furqon, M. `Ariful. (2023). Virtual Assistant for Thesis Technical Guide Using Artificial Neural Network. *Indonesian Journal of Artificial Intelligence and Data Mining*, *6*(2), 188. https://doi.org/10.24014/ijaidm.v6i2.23473

Santoso, D. P., & Wibowo, W. (2022). Analisis Sentimen Ulasan Aplikasi Buzzbreak Menggunakan Metode Naïve Bayes Classifier pada Situs Google Play Store. *Jurnal Sains Dan Seni ITS*, *11*(2). https://doi.org/10.12962/j23373520.v11i2.72534

Sari, H., Ginting, G. L., & Zebua, T. (2021). Penerapan Algoritma Text Mining dan TF-IDF untuk Pengelompokan Topik Skripsi pada Aplikasi Repository STMIK Budi Darma. *TIN: Terapan Informatika Nusantara*, *2*(7), 414–432.

Sheikh, H. A., & Jaiswal, J. (2020). Implementing Sentiment Analysis on Real-Time Twitter Data. *Journal of Emerging Technologies and Innovative Research*, *7*(9), null. https://www.semanticscholar.org/paper/b4f7f1c10077cbf54c5d9f7e5d160aa3cf97c071

Shruthi, P., & Anil Kumar, K. M. (2020). Novel approach for generating hybrid features set to effectively identify hate speech. *Inteligencia Artificial*, *23*(66), 97–111. https://doi.org/10.4114/intartif.vol23iss66pp97-111

Shweta Pandey, Rohit Agarwal, Sachin Bhardwaj, Sanjay Kumar Singh, Dr. Yusuf Perwej, & Niraj Kumar Singh. (2023). A Review of Current Perspective and Propensity in Reinforcement Learning (RL) in an Orderly Manner. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *3307*, 206–227. https://doi.org/10.32628/cseit2390147

Siddiqui, U., Ayaz, S., Nadeem, M., Javed, S., & Saquib Sohail, S. (2021). Characterization of Dataset Exploiting Sentiment Analysis over Twitter. *New Frontiers in Communication and Intelligent Systems*, 215–226. https://doi.org/10.52458/978-81-95502-00-4-24

Taghizadeh-Mehrjardi, R., Schmidt, K., Eftekhari, K., Behrens, T., Jamshidi, M., Davatgar, N., Toomanian, N., & Scholten, T. (2020). Synthetic resampling strategies and machine learning for digital soil mapping in Iran. *European Journal of Soil Science*, *71*(3), 352–368. https://doi.org/10.1111/ejss.12893

Tempola, F., Rosihan, R., & Adawiyah, R. (2021). Holdout Validation for Comparison Classfication Naïve Bayes and KNN of Recipient Kartu Indonesia Pintar. *IOP Conference Series: Materials Science and Engineering*, *1125*(1), 012041. https://doi.org/10.1088/1757-899x/1125/1/012041

Utomo, M. C. C., Taukhid, M., & Mujahidin, S. (2023). Analisis Sentimen Media Sosial Twitter pada Kasus Pemberlakuan Pembatasan Kegiatan Masyarakat dengan menggunakan Metode Naïve Bayes Classifier. *Journal of Mathematics & Information Technology*, *1*(1), 41–48. https://journal.itk.ac.id/index.php/equiva/article/view/815/433

Vindua, R., & Zailani, A. U. (2023). Analisis Sentimen Pemilu Indonesia Tahun 2024 Dari Media Sosial Twitter Menggunakan Python. *JURIKOM (Jurnal Riset Komputer)*, *10*(2), 479. https://doi.org/10.30865/jurikom.v10i2.5945

Wang, C., Han, B., Patel, B., & Rudin, C. (2023). In Pursuit of Interpretable, Fair and Accurate Machine Learning for Criminal Recidivism Prediction. *Journal of Quantitative Criminology*, *39*(2), 519–581. https://doi.org/10.1007/s10940-022-09545-w

Wulandari, D. A., Rohmat Saedudin, R., & Andreswari, R. (2021). Analisis Sentimen Media Sosial Twitter Terhadap Reaksi Masyarakat Pada Ruu Cipta Kerja Menggunakan Metode Klasifikasi Algoritma Naive Bayes. *E-Proceeding of Engineering*, *8*(5), 9007–9016.

You, H., Zhou, M., Zhang, J., Peng, W., & Sun, C. (2023). Sugarcane nitrogen nutrition estimation with digital images and machine learning methods. *Scientific Reports*, *13*(1), 1–12. https://doi.org/10.1038/s41598-023-42190-2

L A M P I R A N

Lampiran 1. Data Mentah

NO	ID	Date	Username	Display Name	Content	URL
1	1.74 E+18	2023-12-28 23:59:44	sujiwo_tego	haha hihi huhu heha	@adriansyahyasin Hei rohingya, ke sini aja... Jakarta. (Pingin lihat respond warga DKI juga)	https://twitter.com/sujiwo_tego/status/1740522953295765692
2	1.74 E+18	2023-12-28 23:59:44	SKoncoro	Koncoro pranoto	@nionaalfaen @neohistoria_id Rohingya itu cuman proyek, kalian paham kan permainan kayak gini gak mungkin asal mau tampung🤣	https://twitter.com/SKoncoro/status/1740522953790661016
3	1.74 E+18	2023-12-28 23:59:24	okvin_mo	Maulindya Okvin	@RistaRL @gojocatty @herricahyadi Itu bentuk kecewa tindak tegas. Emang anak2 & wanita rohingya di pukuli di perkosa mahasiswa??? Mrk ke malaisya awalnya jg jual Tangis🤣.. Udah di Terima malah yg local kalah.. Wataknya beda jauh sm org Palestine.. Dan mrk BUKAN Pengungsi myanmar tp Cari kerja (IMIGRAN GELAP)	https://twitter.com/okvin_mo/status/1740522871825518987
⋮	…	…	…	…	…	…
14135	1.74 E+18	2023-12-25 00:03:55	Kopi_and_ Chat	Jacks Hunters	@folkshittmedia Nanti kalau meninggal datang tuh CNN dan BBC amerika dan membuat statement indonesia menyiksa pengungsi rohingya 😂	https://twitter.com/Kopi_and_Chat/status/1739074455044006169

Seluruh data mentah yang digunakan dalam penelitian ini dapat diakses melalui tautan berikut https://bit.ly/DataMentahRohingya

Lampiran 2. *Syntax Text Preprocessing*

```
#Case Folding
df['Case Folding'] = df['Content'].str.lower()

#Cleaning
import re
import string
import nltk
def remove_URL(tweet): #Menghapus URL/Link
  url = re.compile(r'https?://\S+|www\.\S+')
  return url.sub(r'', tweet)
def remove_emoji(tweet): #Menghapus emoji
  emoji_pattern = re.compile("["
    u"\U0001F600-\U0001F64F"
    u"\U0001F300-\U0001F5FF"
    u"\U0001F680-\U0001F6FF"
    u"\U0001F1E0-\U0001F1FF"
    u"\U000024C2-\U0001F251"
    u"\U0001f926-\U0001f937"
    u"\U00010000-\U0010ffff"
    u"\u2600-\u2B55"
    u"\u200d"
    u"\u23cf"
    u"\u23e9"
    u"\u231a"
    u"\ufe0f"
    u"\u3030"
                      "]+", flags=re.UNICODE)
  return emoji_pattern.sub(r'', tweet)
def remove_angka_dll(tweet):
  tweet = re.sub(r'\d+', '', tweet) #Menghapus angka
  tweet = re.sub(r'\$\w*', '', tweet) #Menghapus ticker pasar saham
  tweet = re.sub(r'^RT[\s]+', '', tweet) #Menghapus RT
  tweet = re.sub(r'#[^\s]+', '', tweet) #Menghapus hashtag
  tweet = re.sub(r'@[^\s]+', '', tweet) #Menghapus mention
  tweet = re.sub(r'\n', ' ', tweet) #Menghapus hal baru
  tweet = re.sub(r'&amp', '', tweet) #Menghapus tanda &amp
  tweet = re.sub(r'[^A-Za-z ]+', ' ', tweet) #Menghapus non alfabet
  tweet = tweet.strip()
  return tweet
df['Cleaning'] = df['Case Folding'].apply(lambda x: remove_URL(x))
df['Cleaning'] = df['Cleaning'].apply(lambda x: remove_emoji(x))
df['Cleaning'] = df['Cleaning'].apply(lambda x: remove_angka_dll(x))

#Normalisasi
kamus_df = pd.read_csv('Kamus Normalisasi Rohingya.csv')
```

```
# Fungsi untuk normalisasi teks berdasarkan kamus normalisasi
def normalisasi(str_text):
    for i in norm:
        if pd.notna(norm[i]):  # Memeriksa apakah nilai normalisasi
tidak NaN
            str_text = str_text.replace(i, norm[i])
        else:
            str_text = str_text.replace(i, '')  # Menghapus kata asal
jika normalisasi adalah NaN
    return str_text
# Membuat kamus normalisasi dari DataFrame kamus_df
norm = dict(zip(kamus_df['Kata Asal'], kamus_df['Kata Normalisasi']))
df['Normalization'] = df['Cleaning'].apply(normalisasi)

#Tokenization
df['Tokenization'] = df['Normalization'].apply(lambda x:x.split())

#Filtering/Stopword Removal
import nltk
from nltk.corpus import stopwords
nltk.download('stopwords')
# Ambil stopwords dalam bahasa Indonesia (atau sesuai kebutuhan)
stop_words = set(stopwords.words('indonesian'))
stop_words.remove('tidak')
# Proses filtering tanpa menghapus kata "tidak"
def remove_stopwords(text):
  return [word for word in text if word not in stop_words]
df['Filtering'] = df['Tokenization'].apply(lambda x:
remove_stopwords(x))

#Steamming Data
!pip install Sastrawi
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
from nltk.stem import PorterStemmer
from nltk.stem.snowball import SnowballStemmer

factory = StemmerFactory()
stemmer = factory.create_stemmer()
def stem_text(text):
  return [stemmer.stem(word) for word in text]
df['Stemming'] = df['Filtering'].apply(lambda x: '
'.join(stem_text(x)))
df.head(15000)
```

Lampiran 3. Hasil *Text Preprocessing*

No	Content	*Case Folding*	*Cleaning*	*Normalization*	*Tokenization*	*Stopword removal*	*Stemming*
1	@adriansyahyasin Hei rohingya, ke sini aja... Jakarta. (Pingin lihat respond warga DKI juga)	@adriansyahyasin hei rohingya, ke sini aja... jakarta. (pingin lihat respond warga dki juga)	hei rohingya ke sini aja jakarta pingin lihat respond warga dki juga	hei rohingya ke sini saja jakarta ingin lihat respon warga dki juga	['hei', 'rohingya', 'ke', 'sini', 'saja', 'jakarta', 'ingin', 'lihat', 'respon', 'warga', 'dki', 'juga']	['hei', 'rohingya', 'jakarta', 'lihat', 'respon', 'warga', 'dki']	hei rohingya jakarta lihat respon warga dki
2	@nionaalfaen @neohistoria_id Rohingya itu cuman proyek, kalian paham kan permainan kayak gini gak mungkin asal mau tampungðŸ¤£	@nionaalfaen @neohistoria_id rohingya itu cuman proyek, kalian paham kan permainan kayak gini gak mungkin asal mau tampungðŸ¤£	rohingya itu cuman proyek kalian paham kan permainan kayak gini gak mungkin asal mau tampung	rohingya itu hanya proyek kalian paham permainan seperti ini tidak mungkin asal mau tampung	['rohingya', 'itu', 'hanya', 'proyek', 'kalian', 'paham', 'permainan', 'seperti', 'ini', 'tidak', 'mungkin', 'asal', 'mau', 'tampung']	['rohingya', 'proyek', 'paham', 'permainan', 'tidak', 'tampung']	rohingya proyek paham main tidak tampung
3	@RistaRL @gojocatty @herricahyadi Itu bentuk kecewa tindak tegas. Emang anak2 & wanita rohingya di pukuli di perkosa mahasiswa???	@ristarl @gojocatty @herricahyadi itu bentuk kecewa tindak tegas. emang anak2 & wanita rohingya di pukuli di perkosa mahasiswa???	itu bentuk kecewa tindak tegas emang anak wanita rohingya di pukuli di perkosa mahasiswa mrk ke malaisya	itu bentuk kecewa tindak tegas memang anak wanita rohingya di pukuli di perkosa mahasiswa mereka ke malaysia	['itu', 'bentuk', 'kecewa', 'tindak', 'tegas', 'memang', 'anak', 'wanita', 'rohingya', 'di', 'pukuli', 'di', 'perkosa', 'mahasiswa', 'mereka', 'ke', 'malaysia',	['bentuk', 'kecewa', 'tindak', 'anak', 'wanita', 'rohingya', 'pukuli', 'perkosa', 'mahasiswa', 'malaysia',	bentuk kecewa tindak anak wanita rohingya pukul perkosa mahasiswa malaysia jual tangis

	Mrk ke malaisya awalnya jg jual TangisðŸ¤£.. Udah di Terima malah yg local kalah.. Wataknya beda jauh sm org Palestine.. Dan mrk BUKAN Pengungsi myanmar tp Cari kerja (IMIGRAN GELAP)	mrk ke malaisya awalnya jg jual tangisðŸ¤£.. udah di terima malah yg local kalah.. wataknya beda jauh sm org palestine.. dan mrk bukan pengungsi myanmar tp cari kerja (imigran gelap)	awalnya jg jual tangis udah di terima malah yg local kalah wataknya beda jauh sm org palestine dan mrk bukan pengungsi myanmar tp cari kerja imigran gelap	awalnya juga jual tangis sudah di terima justru yang lokal kalah wataknya beda jauh sama orang palestina dan mereka bukan pengungsi myanmar tapi cari kerja imigran gelap	'awalnya', 'juga', 'jual', 'tangis', 'sudah', 'di', 'terima', 'justru', 'yang', 'lokal', 'kalah', 'wataknya', 'beda', 'jauh', 'sama', 'orang', 'palestina', 'dan', 'mereka', 'bukan', 'pengungsi', 'myanmar', 'tapi', 'cari', 'kerja', 'imigran', 'gelap']	'jual', 'tangis', 'terima', 'lokal', 'kalah', 'wataknya', 'beda', 'orang', 'palestina', 'pengungsi', 'myanmar', 'cari', 'kerja', 'imigran', 'gelap']	terima lokal kalah watak beda orang palestina ungsi myanmar cari kerja imigran gelap
⋮	…	…	…	…	…	…	…
14135	@folkshittmedia Nanti kalau meninggal datang tuh CNN dan BBC amerika dan membuat statement indonesia menyiksa pengungsi rohingya ðŸ˜,	@folkshittmedia nanti kalau meninggal datang tuh cnn dan bbc amerika dan membuat statement indonesia menyiksa pengungsi rohingya ðŸ˜,	nanti kalau meninggal datang tuh cnn dan bbc amerika dan membuat statement indonesia menyiksa pengungsi rohingya	nanti kalau meninggal datang cnn dan bbc amerika dan membuat pernyatn indonesia menyiksa pengungsi rohingya	['nanti', 'kalau', 'meninggal', 'datang', 'cnn', 'dan', 'bbc', 'amerika', 'dan', 'membuat', 'pernyatn', 'indonesia', 'menyiksa', 'pengungsi', 'rohingya']	['meninggal', 'cnn', 'bbc', 'amerika', 'pernyatn', 'indonesia', 'menyiksa', 'pengungsi', 'rohingya']	tinggal cnn bbc amerika pernyatn indonesia siksa ungsi rohingya

Seluruh data hasil *preprocessing* dalam penelitian ini dapat diakses melalui tautan berikut https://bit.ly/*Preprocessing*Rohingya

Lampiran 4. *Syntax* Pelabelan Data

```
#Translate Data
!pip3 install googletrans==3.1.0a0
from googletrans import Translator
# Inisialisasi objek Translator
translator = Translator()
# Fungsi untuk menerjemahkan teks
def translate_text(text, target_language='en'):
    translation = translator.translate(text, dest=target_language)
    return translation.text
data['Translate'] = data['Stemming'].apply(translate_text)

#Labelling Data
!pip install tweet-preprocessor
!pip install textblob
!pip install wordcloud
!pip install nltk
import preprocessor as p
from textblob import TextBlob
import nltk
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize
nltk.download('punkt')
from textblob import TextBlob
# Fungsi untuk melakukan labelling data menggunakan TextBlob
def label_data(text):
    analysis = TextBlob(text)
    if analysis.sentiment.polarity > 0:
        return 'Positif'
    elif analysis.sentiment.polarity < 0:
        return 'Negatif'
    else:
        return 'Netral'
# Melakukan labelling pada setiap data teks
data['Label'] = data['Translate'].apply(label_data)
# Menghitung jumlah data dengan sentimen positif, negatif, dan netral
total_positif = len(data[data['Label'] == 'Positif'])
total_negatif = len(data[data['Label'] == 'Negatif'])
total_netral = len(data[data['Label'] == 'Netral'])
total = len(data)
# Menampilkan hasil analisis data
print(f'Hasil Analisis Data:\nPositif = {total_positif}\nNegatif =
{total_negatif}\nNetral = {total_netral}')
print(f'\nTotal Data : {total}')
data.head(15000)
```

Lampiran 5. Hasil Pelabelan Data

No.	***Stemming***	***Translate***	**Label**
1	hei rohingya jakarta lihat respon warga dki	*hey rohingya jakarta look at the response of dki residents*	Netral
2	rohingya proyek paham main tidak tampung	*rohingya project understands that the game cannot be accommodated*	Negatif
3	bentuk kecewa tindak anak wanita rohingya pukul perkosa mahasiswa malaysia jual tangis terima lokal kalah watak beda orang palestina ungsi myanmar cari kerja imigran gelap	*a form of disappointment in the actions of a rohingya woman who was beaten and raped by a malaysian student selling tears accepting that local people are losing different characters palestinians fleeing myanmar looking for work illegal immigrants*	Negatif
4	sadar komen hati nurani postingan rohingya	*be aware of the conscience comments on rohingya posts*	Positif
5	gak mental orang rohingya kenal uang	*rohingya people dont think they know money*	Netral
6	timpang tepis dicounter speak isu ramai fokus mini jihad online palestina time speak rohingya luas literasi luas rizki yuk melek hati luas empati	*lame edge on counter speak busy issue focus mini jihad online palestine time speak rohingya wide literacy wide fortune lets be literate wide heart empathy*	Negatif
7	tulis singkat rohingya	*write short rohingya*	Netral
8	pikir ih tidak suka rohingya beban masyarakat baca baca diperhatiin gapantas usir benci oknum umanize lagisg	*i think i dont like rohingya its a burden on society to read pay attention its okay to get rid of hate people who are umanize again*	Negatif
9	sejarah rohingya	*rohingya history*	Netral
10	ungsi rohingya warga aceh kabur negara dom	*rohingya residents of aceh fled the country*	Netral
⋮	…	…	…
14117	tinggal cnn bbc amerika pernyatn indonesia siksa ungsi rohingya	*all that remains is cnn bbc americas statement that indonesia is torturing rohingya refugees*	Netral

Seluruh data hasil pelabelan dalam penelitian ini dapat diakses melalui tautan berikut https://bit.ly/PelabelanRohingya

Lampiran 6. *Syntax* pembagian data

```
from sklearn.model_selection import train_test_split

#Membagi dataset menjadi data latih dan data uji
X_train, X_test, y_train, y_test = train_test_split(data['Translate'],
data['Label'], test_size=0.2, random_state=42)
print(f'Jumlah Data Latih: {len(X_train)}')
print(f'Jumlah Data Uji: {len(X_test)}')
```

Lampiran 7. *Syntax* pembobotan kata dengan TF-IDF

```
from sklearn.feature_extraction.text import TfidfVectorizer

# Membuat objek TfidfVectorizer
tfidf_vectorizer = TfidfVectorizer()
# Menghitung TF-IDF dari data latih dan uji
X_train_tfidf = tfidf_vectorizer.fit_transform(X_train)
X_test_tfidf = tfidf_vectorizer.transform(X_test)

# Mendapatkan daftar term
terms = tfidf_vectorizer.get_feature_names_out()
# Mendapatkan nilai IDF
idf_values = tfidf_vectorizer.idf_
# Menampilkan hasil TF-IDF
print("Term Frequency (TF):\n", terms)
print("\nInverse Document Frequency (IDF):\n", idf_values)
print("\nTF-IDF:\n", X_train_tfidf)
```

Lampiran 8. *Syntax* Penyeimbangan Kelas Data

```
from imblearn.over_sampling import SMOTE

# Menggunakan SMOTE untuk oversampling
smote = SMOTE(random_state=42)
X_train_resampled, y_train_resampled =
smote.fit_resample(X_train_tfidf, y_train)
```

Lampiran 9. *Syntax Tuning Hyperparameter*

```
from sklearn.model_selection import GridSearchCV

# Untuk SVM
svm_parameters = {'kernel': ['linear', 'rbf', 'poly', 'sigmoid'], 'C':
[0.1, 1, 10], 'random_state': [0, 42, 100]}
svm = SVC(probability=True)
svm_grid_search = GridSearchCV(svm, svm_parameters, cv=5,
scoring='accuracy')
svm_grid_search.fit(X_train_resampled, y_train_resampled)
print(f'Best SVM Params: {svm_grid_search.best_params_}')
print(f'Best SVM Cross-Validation Score:
{svm_grid_search.best_score_}')

# Untuk Gradient Boosting
gb_parameters = {'n_estimators': [100, 200, 300], 'learning_rate':
[0.01, 0.1, 1.0], 'max_depth': [1, 2, 3, 4, 5], 'random_state': [0, 42,
100]}
gb = GradientBoostingClassifier()
gb_grid_search = GridSearchCV(gb, gb_parameters, cv=5,
scoring='accuracy')
gb_grid_search.fit(X_train_resampled, y_train_resampled)
print(f'Best GB Params: {gb_grid_search.best_params_}')
print(f'Best GB Cross-Validation Score: {gb_grid_search.best_score_}')
```

Lampiran 10. *Syntax* Klasifikasi Data (*Support Vector Machine*)

```
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt

#Inisialisasi model SVM
svm_model =SVC(kernel='linear', C=1.0, random_state=0,
probability=True)
svm_model.fit(X_train_resampled, y_train_resampled)
# Menampilkan hyperparameter model
print('Hyperparameter model SVM:')
print(f'kernel: {svm_model.kernel}')
print(f'C: {svm_model.C}')
print(f'intercept: {svm_model.intercept_}')
print(f'coef_: {svm_model.coef_}')

# Evaluasi model menggunakan data uji
y_pred_prob_svm = svm_model.predict(X_test_tfidf)
# Akurasi pada data uji
accuracy_svm = accuracy_score(y_test, y_pred_prob_svm)
print(f'Accuracy on Testing Data: {accuracy_svm*100:.2f}%')
# Menampilkan classification report
print('\nClassification Report Support Vector Machine:\n',
classification_report(y_test, y_pred_prob_svm))
print('\nConfusion Matrix:\n', confusion_matrix(y_test,
y_pred_prob_svm))

# Melatih model menggunakan data latih
predictions_train = svm_model.predict(X_train_resampled)
# Menggabungkan hasil prediksi dari data uji dan data latih
Klasifikasi_svm = pd.concat([pd.Series(predictions_train),
pd.Series(y_pred_prob_svm)], ignore_index=True)
# Menambahkan kolom hasil prediksi Naive Bayes ke dalam DataFrame data
data['Klasifikasi SVM'] = Klasifikasi_svm
data.head()
```

Lampiran 11. *Syntax* Klasifikasi Data (*Gradient Boosting*)

```
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt

# Membuat model Gradient Boosting
gb_classifier = GradientBoostingClassifier(n_estimators=300,
learning_rate=1.0, max_depth=3, random_state=0)
# Melatih model menggunakan data latih
gb_classifier.fit(X_train_resampled, y_train_resampled)

# Melakukan prediksi pada data uji
y_pred_prob_gb = gb_classifier.predict(X_test_tfidf)
# Mengukur akurasi model
accuracy_gb = accuracy_score(y_test, y_pred_prob_gb)
print(f"Akurasi Gradient Boosting: {accuracy_gb*100:2f}%")
# Menampilkan classification report
print('\nClassification Report Gradient Boosting:\n',
classification_report(y_test, y_pred_prob_gb))
print('\nConfusion Matrix:\n', confusion_matrix(y_test,
y_pred_prob_gb))

# Evaluasi model menggunakan data latih
predictions_train = gb_classifier.predict(X_train_resampled)

# Menggabungkan hasil prediksi dari data uji dan data latih
Klasifikasi_gb = pd.concat([pd.Series(predictions_train),
pd.Series(y_pred_prob_gb)], ignore_index=True)

# Menambahkan kolom hasil prediksi Naive Bayes ke dalam DataFrame data
data['Klasifikasi GB'] = Klasifikasi_gb
data.head()
```

Lampiran 12. *Syntax Cross Validation*

```
from sklearn.model_selection import cross_val_score
import numpy as np

# Lakukan cross-validation dengan 5 fold
cv_scores = cross_val_score(gb_classifier, X_train_resampled,
y_train_resampled, cv=5, scoring='accuracy')

# Tampilkan hasil cross-validation
print(f'Cross-Validation Scores: {cv_scores}')
print(f'Mean Cross-Validation Score: {cv_scores.mean()}')
print(f'Standard Deviation: {cv_scores.std()}')
```

Lampiran 13. Hasil Klasifikasi *Support Vector Machine* dan *Gradient Boosting*

No.	*Stemming*	Translate	Label	Klasifikasi SVM	Klasifikasi GB
1	hei rohingya jakarta lihat respon warga dki	*hey rohingya jakarta look at the response of dki residents*	Netral	Positif	Positif
2	rohingya proyek paham main tidak tampung	*rohingya project understands that the game cannot be accommodated*	Negatif	Negatif	Negatif
3	bentuk kecewa tindak anak wanita rohingya pukul perkosa mahasiswa malaysia jual tangis terima lokal kalah watak beda orang palestina ungsi myanmar cari kerja imigran gelap	*a form of disappointment in the actions of a rohingya woman who was beaten and raped by a malaysian student selling tears accepting that local people are losing different characters palestinians fleeing myanmar looking for work illegal immigrants*	Negatif	Netral	Netral
4	sadar komen hati nurani postingan rohingya	*be aware of the conscience comments on rohingya posts*	Positif	Positif	Netral
5	gak mental orang rohingya kenal uang	*rohingya people dont think they know money*	Netral	Positif	Positif
⋮	…	…	…	…	…
14117	tinggal cnn bbc amerika pernyatn indonesia siksa ungsi rohingya	*all that remains is cnn bbc americas statement that indonesia is torturing rohingya refugees*	Netral	Positif	Positif

Seluruh data hasil klasifikasi *Support Vector Machine* dan *Gradient Boosting* dalam penelitian ini dapat diakses melalui tautan berikut https://bit.ly/HasilAnalisisSentimenRohingya