Klasifikasi Sentimen Menggunakan Algoritma ¬K-Nearest Neighbour (Studi Kasus: Magang Merdeka Belajar)

Hollywrit Travaganz Nainggolan[1], Bayu Hananto[2], Bambang Tri Wahyono[3]
[1,2,3] Informatika, Fakultas Ilmu Komputer
[1,2,3] Universitas Pembangunan Nasional Veteran Jakarta
[1,2,3] Jl. RS Fatmawati No. 1, Pondok Labu, Jakarta Selatan DKI Jakarta 12450
beezanteeum@upnvj.ac.id [1], bayuhananto@upnvj.ac.id*[2], bambang.triwahyono@upnvj.ac.id [3]

Abstrak. Pada tahun 2020, Kementerian Pendidikan dan Kebudayaan meluncurkan program Merdeka Belajar yang membantu para mahasiswa dan mahasiswi untuk menghadapi lingkungan kerja setelah mereka lulus. Akan tetapi, program ini memunculkan polemik. Penelitian ini mencoba melakukan klasifikasi sentimen pada kasus Magang Merdeka Belajar menggunakan algoritma KNN. KNN dipilih dikarenakan algoritma ini lebih handal dalam menangani data noisy, namun untuk meningkatkan akurasi, peneliti menggunakan algoritma Near Miss dalam proses data balancing dikarenakan selisih data adalah 188 data positif dan 212 data negatif.

Kata Kunci: Klasifikasi Sentimen, KNN, Magang Merdeka, Kampus Merdeka, MBKM

1 Pendahuluan

Pada tahun 2020, Pemerintah melalui Kementerian Pendidikan dan Kebudayaan meluncurkan program Merdeka Belajar Kampus Merdeka (topik yang akan dibahas di sini adalah magang). Akan tetapi, program kerja ini menuai polemik di antara para peserta.

Penelitian ini dilaksanakan untuk melakukan klasifikasi sentimen terhadap program kerja Magang Merdeka Belajar menggunakan algoritma KNN dan *Near Miss* untuk melakukan *data balancing*. Penggunaan algoritma KNN dipilih karena penggunaan pada kasus klasifikasi sentimen cukup banyak, seperti pada kasus Pilkada DKI Jakarta [1], kepuasan konsumen layanan logistik [2], dan kepuasan konsumen layanan travel [3], serta algoritma KNN ini *reliable* bahkan untuk data yang noisy sekalipun. Penelitian ini bertujuan untuk mengetahui klasifikasi sentimen pada kasus Magang Merdeka belajar apakah positif atau negatif.

2 Tinjauan Pustaka

2.2 Magang Merdeka Belajar

Merdeka Belajar (lebih spesifiknya: Kampus Merdeka) adalah program kerja yang diselenggarakan oleh Kementerian Pendidikan dan Kebudayaan dalam rangka membantu mahasiswa dalam menghadapi persaingan di dunia kerja. Program kerja ini sesuai dengan Permendikbud Nomor 3 tahun 2020 pasal 18 tentang Standar Nasional Pendidikan Tinggi.[4]

2.3 Klasifikasi Sentimen

Klasifikasi Sentimen adalah proses untuk melakukan analisis perasaan dan opini, lalu melakukan evaluasi dari hasil analisis tersebut.[5]

2.4 *Text Mining*

Text Mining adalah proses pengolahan pengetahuan intensif dimana pengguna mengolah sekumpulan dokumen dari hari-ke-hari menggunakan peralatan analisis, sehingga dapat menjadi jawaban pada pemrosesan pada data yang tak terstruktur.[6]

2.5 Pembobotan TF-IDF

Algoritma TF-IDF (Term frequency-inverse document frequency) adalah algoritma yang digunakan untuk menghitung bobot setiap kata yang paling umum digunakan pada *information retrieval*. Diharapkan dengan algoritma ini dapat menambah tingkat keefektifan hasil kemiripan dan waktu yang dibutuhkan untuk melakukan proses pendeteksian.[7]

Penghitungan TF-IDF menggunakan rumus sebagai berikut:

$$w = tf \times idf \quad (1)$$

Sedangkan IDF sendiri dicari menggunakan rumus sebagai berikut:

$$idf = \log \frac{D}{Df_i} \quad (2)$$

Dimana:
w = bobot dokumen
tf = jumlah kata yang dicari dalam dokumen
idf = Inverse Document Frequency, yaitu hubungan antara kata dengan dokumen.

2.6 Algoritma *Near Miss*

Algoritma *Near Miss* adalah metode undersampling berupa pengembangan dari KNN, dimana data mayoritas diambil dan dilakukan proses *undersampling* sehingga jumlah data mayoritas dan minoritas menjadi sama.[8]

2.7 *K-Nearest Neighbor*

Algoritms K-Nearest Neighbor adalah algoritma yang menggunakan prinsip untuk mengambil K data tetangga untuk proses pembagian kelas. KNN membagi data berdasar penelitian terhadap data lainnya.[9]

2.8 *Confusion Matrix*

Confusion Matrix adalah diagram untuk mengukur kinerja suatu algoritma pada model machine learning dimana diagram ini membantu untuk menghitung berapa banyak data yang dikenali secara tepat atau keliru. Pengukuran confusion matrix antara lain True Positive, False Positive, True Negative, dan False Negative. Dan dari keempat nilai tersebut, kita dapat mencari precision, accuracy, specificity, dan recall, dimana rumusnya adalah:

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative} \quad (3)$$

$$Precision_{positive} = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (4)$$

$$Precision_{negative} = \frac{True\ Negative}{True\ Negative + False\ Negative} \quad (5)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (6)$$

$$Specificity = \frac{True\ Negative}{True\ Negative + False\ Positive} \quad (7)$$

Dimana:

True Positive	: Data sesungguhnya dan data prediksi sama-sama positif
False Positive	: Data prediksi positif, tapi data sesungguhnya negatif
True Negative	: Data sesungguhnya dan data prediksi sama-sama negatif
False Negative	: Data prediksi negatif, tapi data sesungguhnya positif

3 Hasil dan Pembahasan

Data diambil sejumlah 400 data dari Twitter menggunakan library Scweet via Python, lalu data yang diambil dilakukan preprocessing menggunakan library built-in Python, NLTK, dan Sastrawi

Kemudian, data tersebut dilakukan pelabelan secara manual oleh tiga annotator, dimana penilaian yang dilakukan bersifat subyektif.

Tabel 1. Contoh data sentimen

Tweet	Sentimen 1	Sentimen 2	Sentimen 3
buat poin yg magang, *i beg to differ*. *kan* magang merdeka belajar ini merupakan hal opsional? artinya ya mau diambil ataupun tidak, hal tsb kembali ke mahasiswa.	Positif	Positif	Positif
Kesempatan buat magang sekarang juga lebih terbuka. Kemendikbud bisa ngepush perusaha2an keren buat bikin program magang via magang merdeka. Ini kebijakan keren sih di era menteri Nadim	Positif	Positif	Positif
sisi lain seneng dapet job nambah fee tapi sok kadang magang blm kelar dah dikejar deadline pen cepet oktober merdeka	Negatif	Negatif	Negatif
Mau daftar kampus Merdeka magang, tapi pas liat persyaratan gaada yg butuh sastra anjgg w daftar kmn dong	Negatif	Negatif	Negatif

Kemudian, dari data tersebut, dicarilah *Kappa value* sesuai dengan rumus (8):

$$\kappa = \frac{p_{\propto} - p_{\varepsilon}}{1 - p_{\varepsilon}} \quad (8)$$

Dimana p_0 dan p_ε masing-masing ditentukan dengan rumus (9) dan (10):

$$p_{\propto} = \frac{1}{n}\sum_{i=1}^{n} q_i \quad (9)$$

$$p_{\varepsilon} = \sum_{j=1}^{n} q_j^2 \quad (10)$$

Dan q_i serta q_j ditentukan dengan rumus (11) dan (12):

$$q_i = \frac{1}{m(m-1)}\sum_{j=1}^{k} q_{ij}^2 - q_{ij} \quad (11)$$

$$q_j = \frac{xj}{nm} = \frac{1}{nm}\sum_{i=1}^{n} x_{ij} \quad (12)$$

Dimana:

p_0 : Persentase pengukuran *annotator*
p_ε : Persentase perubahan *annotator*
n : Jumlah data
m : Jumlah *annotator*
$x_{ij} = \sum_{i=1}^{n} x_{ij}$: Total label setiap kategori

2.2 Pencarian *Kappa Value*

Dari rumus (8), (9), (10), (11), dan (12), maka didapatkan *Kappa Value* sebagai berikut:
Jumlah q_i adalah:

$$q_1 = \frac{1}{3(3-1)} \times (3^2 + 0^2 - 3)$$
$$= \frac{1}{6} \times 6$$
$$= 1$$
$$q_2 = \frac{1}{3(3-1)} \times (0^2 + 3^2 - 3)$$
$$= \frac{1}{6} \times 6$$
$$= 1$$
$$q_3 = \frac{1}{3(3-1)} \times (2^2 + 1^2 - 3)$$
$$= \frac{1}{6} \times 2$$
$$= 0{,}333$$
$$\sum_{j=1}^{N} q_n = q_1 + q_2 + q_3 + q_4 + \cdots + q_N$$
$$= 1 + 1 + 0{,}333 + \cdots$$
$$= 390{,}667$$

Sedangkan q_j adalah 751 data positif dan 835 data negatif. Kemudian, nilai p_ε didapatkan dengan rumus berikut:

$$P_{positif} = \frac{1}{400} \times 751$$
$$= 0{,}626$$
$$P_{negatif} = \frac{1}{400} \times 835$$
$$= 0{,}696$$
$$p_\varepsilon = 0{,}626^2 + 0{,}696^2$$
$$= 0{,}876$$

Dari data di atas, maka *Kappa Value* adalah:

$$\kappa = \frac{0{,}977 - 0{,}876}{1 - 0{,}876}$$
$$= \frac{0{,}101}{0{,}188}$$
$$= 0{,}812$$

Karena *Kappa value* yang didapatkan adalah 0,812 maka penelitian dapat dilanjutkan

2.3 *Case Folding*

Pada tahapan ini, kapitalisasi pada corpus diubah menjadi huruf kecil, dikarenakan kapitalisasi pada corpus dapat mempengaruhi proses pengolahan sentimen.

```
import pandas
data = read csv using pandas
data['tweets'] = apply lower() function to data['Tweets']
```

2.4 Pembersihan Data

Pembersihan data dilakukan untuk membuang karakter yang tidak diperlukan, seperti angka, tanda baca, spasi berganda, dan lain-lain.

```
import string
import re
```

```
create function to replace enters, tabs, single characters, numbers, punctuations,
double space, and whitespace with empty string
data['Tweets']=apply function to tweet
```

2.5 *Tokenisasi*

Tokenisasi dilakukan untuk membagi corpus menjadi sekumpulan token, yaitu penggalan kata-kata dari setiap corpus.

```
import word_tokenize from NLTK
create function to tokenize tweets by using word_tokenize
data['token']=apply tokenize function to data['Tweets']
```

2.6 *Filtering*

Setelah melakukan tokenisasi, maka dilanjutkan ke tahap filtering, yaitu melakukan pembuangan dari stopword.

```
import NLTK
download stopwords corpus from NLTK
import stopwords from NLTK
list_stopwords = stopwords.words('indonesian')
extend list_stopwords with neccesary words
removed = []←create removed array
if word in data['tweets'] not included in list_stopwords:
             append words to removed[] array
data['no_stopwords'] = removed
```

2.7 *Stemming*

Setelah proses filtering, maka proses selanjutnya adalah stemming, yaitu mengubah kata yang berimbuhan menjadi kata dasar.

```
import StemmerFactory from Sastrawi
import swifter
create factory from StemmerFactory()
term_dict{}←create term_dict array
create stemming function
data['stemmed']=apply swifter stemming function to data['no_stopword']
```

2.8 *Normalisasi*

Token yang sudah dijadikan kata dasar kemudian dinormalisasi. Proses normalisasi adalah proses untuk perbaikan kesalahan tik atau kata tidak baku di dalam suatu *corpus* atau token.

```
normalize=read_excel from normalization excel file using pandas
normalize_dict = [] ← create normalization array
for index, row in normalize.iterrows():
             if row[0] not in normalize dictionary table:
                    normalize table[row[0]] = row[1]
normalized_term function: apply normalize table if term in normalize table
data['normalized'] = apply normalized_term function to stemmed token
```

2.9 Pencarian *Kappa Value*

Setelah data di-praproses, data ini kemudian dihitung bobotnya menggunakan TF-IDF. Pada kasus ini, peneliti menggunakan empat sampel data.

Tabel 2. Sampel data latih

No	Token	Sentimen
1	'magang', 'sertifikat', 'kampus', 'merdeka', 'program', 'magang', 'cepat', 'akselerasi', 'alam', 'ajar', 'rancang', 'baik'	Positif
2	'sempat', 'magang', 'buka', 'kemendikbud', 'paksa', 'usaha', 'keren', 'buat', 'buat', 'program', 'magang', 'magang', 'merdeka', 'bijak', 'keren', 'menteri'	Positif
3	'sesal', 'dahulu', 'tidak', 'ikut', 'daftar', 'magang', 'kampus', 'merdeka', 'awal', 'awal', 'gagal', 'magang', 'rintis', 'didik', 'lihat', 'berapa', 'utas', 'sedikit', 'buka', 'mata', 'syukur', 'tidak', 'sempat', 'rasa'	Negatif
4	'daftar', 'kampus', 'merdeka', 'magang', 'saat', 'lihat', 'syarat', 'tiada', 'butuh', 'sastra', 'daftar', 'mana'	Negatif

Dari data latih tersebut, kemudian dihitung menggunakan rumus (1) dan (2), sehingga hasilnya adalah sebagai berikut.

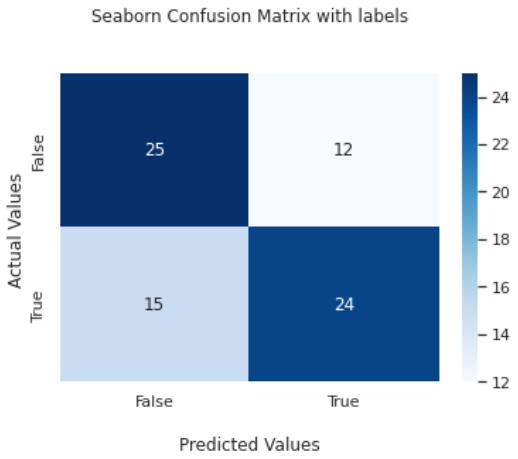
Tabel 3. Pembobotan TF-IDF

Kata	TF				Jumlah TF	IDF	TF-IDF
	D1	D2	D3	D4			
magang	2	2	2	1	7	-0,243	-1,701
sertifikat	1	0	0	0	1	0,602	0,602
kampus	1	0	1	1	3	0,125	0,375
merdeka	1	1	1	1	4	0,000	0,000
cepat	1	0	0	0	1	0,602	0,602
akselerasi	1	0	0	0	1	0,602	0,602
alam	1	0	0	0	1	0,602	0,602
ajar	1	0	0	0	1	0,602	0,602
rancang	1	0	0	0	1	0,602	0,602
baik	1	0	0	0	1	0,602	0,602
sempat	0	1	1	0	2	0,301	0,602
buka	0	1	1	0	2	0,301	0,602
kemendikbud	0	1	0	0	1	0,602	0,602
paksa	0	1	0	0	1	0,602	0,602
usaha	0	1	0	0	1	0,602	0,602
keren	0	2	0	0	2	0,301	0,602
buat	0	2	0	0	2	0,301	0,602
program	0	1	0	0	1	0,602	0,602
bijak	0	1	0	0	1	0,602	0,602
menteri	0	1	0	0	1	0,602	0,602
sesal	0	0	1	0	1	0,602	0,602
dahulu	0	0	1	0	1	0,602	0,602
tidak	0	0	2	0	2	0,301	0,602
ikut	0	0	1	0	1	0,602	0,602
daftar	0	0	1	2	3	0,125	0,375
awal	0	0	2	0	2	0,301	0,602
gagal	0	0	1	0	1	0,602	0,602
rintis	0	0	1	0	1	0,602	0,602
didik	0	0	1	0	1	0,602	0,602

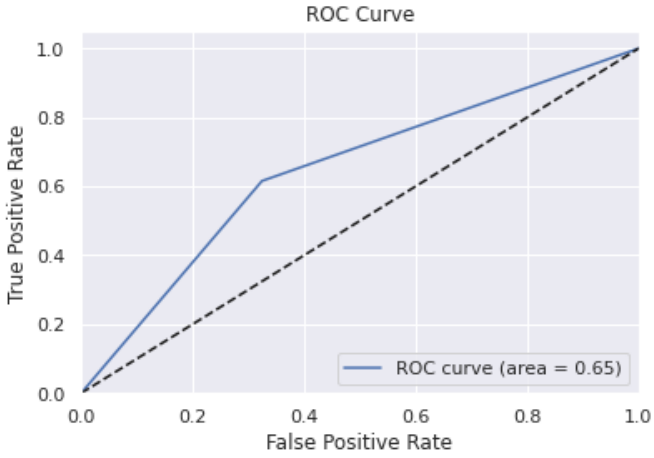
ISSN: 2655-139X (ONLINE)
ISSN: 0216-4221 (PRINT)

lihat	0	0	1	1	2	0,301	0,602
berapa	0	0	1	0	1	0,602	0,602
utas	0	0	1	0	1	0,602	0,602
sedikit	0	0	1	0	1	0,602	0,602
mata	0	0	1	0	1	0,602	0,602
syukur	0	0	1	0	1	0,602	0,602
rasa	0	0	1	0	1	0,602	0,602
saat	0	0	0	1	1	0,602	0,602
syarat	0	0	0	1	1	0,602	0,602
tiada	0	0	0	1	1	0,602	0,602
butuh	0	0	0	1	1	0,602	0,602
sastra	0	0	0	1	1	0,602	0,602
mana	0	0	0	1	1	0,602	0,602

Adapun proses klasifikasi adalah menggunakan algoritma KNN, dimana data sebelumnya dilakukan balancing menggunakan algoritma Near Miss sebelum proses klasifikasi. Setelah proses balancing, data diklasifikasi dengan pembagian data 80% data latih dan 20% data uji dengan n = 3. Hasil dari Confusion matrix dan ROC adalah seperti dicontohkan pada Gambar 1 dan Gambar 2.



Gambar. 1. *Confusion Matrix*. Terlihat bahwa TN = 25, FN = 15, FP = 12, dan TP = 24



Gambar. 2. Kurva ROC

ISSN: 2655-139X (ONLINE)
ISSN: 0216-4221 (PRINT)

Dari confusion matrix yang ada di gambar 1, dapat dicari accuracy, precision, recall, dan specificity sesuai dengan rumus (3), (4), (5), (6), (7).

$$Accuracy = \frac{24 + 25}{24 + 25 + 12 + 15}$$
$$= 0{,}645$$
$$Precision_{pos} = \frac{24}{24 + 12}$$
$$= 0{,}667$$
$$Precision_{neg} = \frac{25}{25 + 15}$$
$$= 0{,}625$$
$$Recall = \frac{24}{24 + 15}$$
$$= 0{,}615$$
$$Specificity = \frac{25}{25 + 12}$$
$$= 0{,}676$$

4 Hasil Pembahasan

1. Pada penelitian ini, sentimen terhadap kasus Magang Merdeka Belajar cenderung positif, dengan 24 data True Positive dan 15 data False Negative.
2. Penelitian ini berjalan cukup baik dikarenakan antara nilai accuracy, precision, recall, dan specificity tidak berbeda jauh. Nilai accuracy setelah dibulatkan adalah 65%, nilai precision poisitive dan precision negative setelah dibulatkan masing-masing adalah 67% dan 63%, nilai recall setelah dibulatkan adalah 62%, sedangkan nilai specificity setelah dibulatkan adalah 68% dengan total data adalah 376 setelah dilakukan proses balancing menggunakan algoritma Near Miss.

Referensi

[1] A. Deviyanto dan M. D. R Wahyudi, "Penerapan Analisis Sentimen pada Pengguna Twitter Menggunakan Metode K-Nearest Neighbor," *Jurnal Informatika Sunan Kalijaga*, vol. 3, no. 1, pp. 1–13, 2018.

[2] A. Salam, J. Zeniarja, R. Septiyan, dan U. Khasanah, "Analisis Sentimen Data Komentar Sosial Media Facebook dengan K-Nearest Neighbor (Studi Kasus pada Akun Jasa Ekspedisi Barang J&T Ekspress Indonesia) ," vol. 2, 2018.

[3] S. Ernawati dan R. Wati, "Penerapan Algoritma K-Nearest Neighbors pada Analisis Sentimen Review Agen Travel," *Jurnal KHATULISTIWA Informatika*, vol. VI, no. 1, pp. 64–69, 2018.

[4] Dirjen Kemendikbud, "Buku Panduan Merdeka Belajar-Kampus Merdeka," Jakarta, Departemen Pendidikan dan Kebudyaan, 2020.

[5] T. Wilson, J. Wiebe, dan P. Hoffmann, "Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis," *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, vol. 1, no. 1, pp. 347–354, 2005 doi: 10.5120/1160-1453.

[6] R. Feldman dan J. Sanger, "The Text Mining Handbook: Advanced Approacehs in Analyzing Unstructured Data," 2005

[7] S. N. Lolyta, R. Y. Dillak, dan F. E. Laumal, 2019. "Sistem Deteksi Plagiarisme Lintas Bahasa Menggunakan Algoritma Tf-Idf," *Jurnal Ilmiah FLASH*, vol. 5, no. 1, pp. 29–32, 2019.

[8] D. Devi, S. K. Biswas, dan B. Purkayastha, 2020. "A Review on Solution to Class Imbalance Problem: Undersampling Approaches," *in International Conference on Computational Performance Evaluation*, pp. 626–631, 2020, doi: 10.1109/ComPE49325.2020.9200087.

[9] L. Afifah, "Algoritma K-Nearest Neighbor (KNN) untuk Klasifikasi," https://ilmudatapy.com/algoritma-k-nearest-neighbor-knn-untuk-klasifikasi/. Diakses November 2021