

Klasifikasi Teks Soal Ujian Berbahasa Indonesia Berdasarkan Ranah Kognitif Taksonomi Bloom

1301174597

Justisio Yan Prawira Adam



1. Pendahuluan

Latar Belakang

- Ujian tulis digunakan untuk mengidentifikasi capaian belajar dari siswa.
- Ujian yang diberikan dapat berisi soal dengan tingkat kesulitan yang berbeda.
- Tingkatan kognitif dari Taksonomi Bloom dapat dijadikan acuan.

Permasalahan

- Waktu untuk melakukan klasifikasi manual, selaras dengan jumlah soal yang akan diklasifikasi.
- Klasifikasi manual rentan akan perbedaan persepsi antar pengajar, akibatnya hasil klasifikasi bisa berbeda.
- Oleh karena itu, Tugas Akhir ini mencoba untuk melakukan klasifikasi secara otomatis.
- Algoritma yang digunakan adalah Support Vector Machine dan Naïve Bayes.
- Karena menghasilkan performa yang baik pada penelitian Patil et al. [5] dan Aninditya et al. [3].



1. Pendahuluan

Batasan Masalah

- Data yang digunakan bersifat tekstual dengan Bahasa Indonesia
- Mata pelajaran yang digunakan adalah Bahasa Indonesia, matematika dan Ilmu Pengetahuan Alam (IPA)
- Klasifikasi hanya bertujuan untuk menentukan tingkatan kognitif yang sesuai dari sebuah soal

Tujuan Penelitian

- Melakukan klasifikasi teks berbahasa Indonesia dengan metode SVM dan NB yang kemudian akan diukur performansi dari masing-masing metode.



2. Studi Terkait

Penulis	Dataset	Metode	Skor
Kusuma et al. [1]	130 Soal Berbahasa Indonesia	Ekstraksi fitur leksikal dan sintaktik Algoritma SVM dengan kernel linear	Rata-rata akurasi 88,6%
Aninditya et al. [2]	Soal ujian semester Berbahasa Indonesia dari Departemen SISFO Universitas Telkom Klasifikasi Biner (2 kelas)	Ekstraksi fitur N-gram TF-IDF Algoritma Naïve Bayes	Precision 85%
Patil et al. [5]	1000 Pertanyaan Berbahasa Inggris	Algoritma SVM dan KNN	Akurasi SVM 0.923 Akurasi KNN 0.666
Mohammed et al. [12]	Pertanyaan terbuka dalam Bahasa Inggris 141 soal dari website 600 soal dari penelitian sebelumnya	Ekstraksi fitur TF-IDF, TFPOS-IDF, W2V-TFPOSIDF. Algoritma SVM, KNN, LR.	SVM, Data penelitian sebelumnya TF-IDF: F1-measure 0.826 TFPOS-IDF: F1-measure 0.866 W2V-TFPOSIDF: F1-measure 0.897



2. Studi Terkait

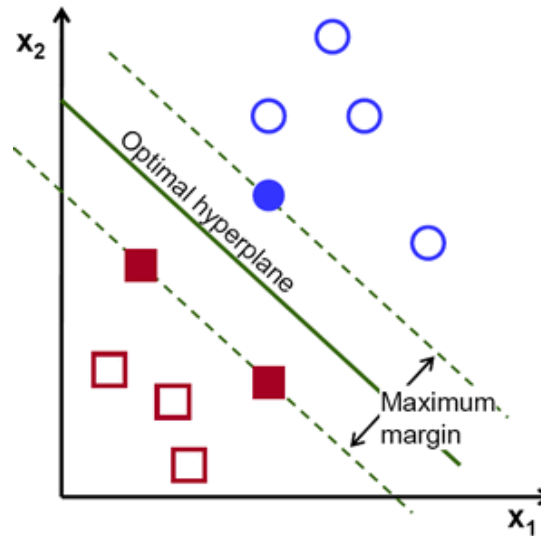
Taksonomi Bloom

Tingkatan Kognitif	Penjelasan
Mencipta (C6)	kemampuan mengevaluasi dan menilai sesuatu berdasarkan norma, acuan atau kriteria.
Mengevaluasi (C5)	kemampuan merangkai atau menyusun kembali komponen - komponen dalam rangka menciptakan arti/pemahaman/ struktur baru
Menganalisa (C4)	kemampuan memisahkan konsep ke dalam beberapa komponen untuk memperoleh pemahaman yang lebih luas atas dampak komponen - komponen terhadap konsep tersebut secara utuh
Menerapkan (C3)	kemampuan menggunakan konsep dalam praktek atau situasi yang baru
Memahami (C2)	kemampuan memahami instruksi/masalah, menginterpretasikan dan menyatakan kembali dengan kata-kata sendiri
Mengingat (C1)	kemampuan menyebutkan atau menjelaskan kembali



2. Studi Terkait

Support Vector Machine



2. Studi Terkait

Naïve Bayes

$$\hat{y} = \arg \max P(y) \prod_{i=1}^n P(x_i | y)$$



2. Studi Terkait

TF-IDF

Memberikan bobot pada setiap kata, yang menunjukkan seberapa penting kata tersebut dalam dokumen.

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$



2. Studi Terkait

TFPOS-IDF

Memberikan pembobotan pada kata berdasarkan POS Tagnya masing-masing.
Dilakukan normalisasi dengan L2 Norm setelah penghitungan TFPOS-IDF.

$$w_{pos}(t) = \begin{cases} w_1 & \text{if } t \text{ is verb} \\ w_2 & \text{if } t \text{ is noun or adjective} \\ w_3 & \text{otherwise} \end{cases}$$

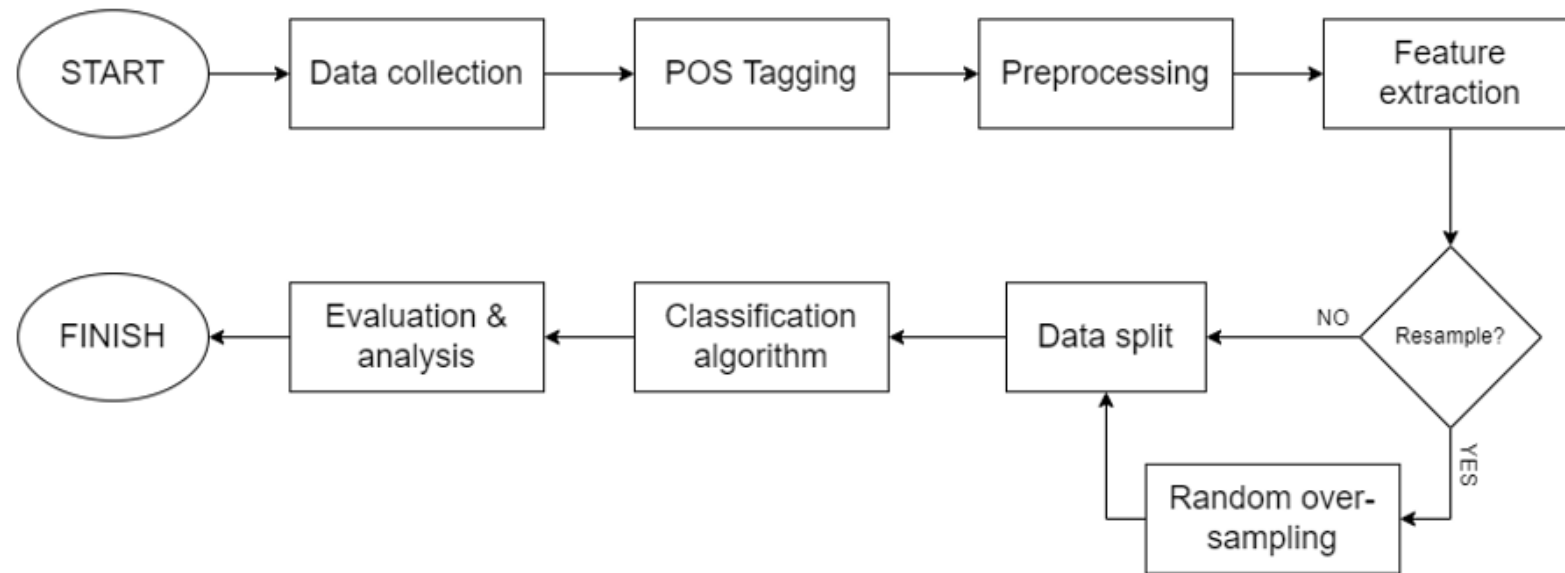
$$w_1 = 5, w_2 = 3, w_3 = 1$$

$$TFPOS(t, d) = \frac{c(t, d) * w_{pos}(t)}{\sum_i c(t_i, d) * w_{pos}(t)}$$

$$TFPOS - IDF(t, d) = TFPOS(t, d).IDF(t)$$

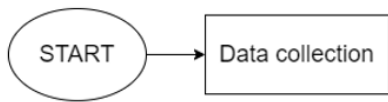


3. Sistem Yang Dibangun



Gambar 2 Alur Kerja Sistem



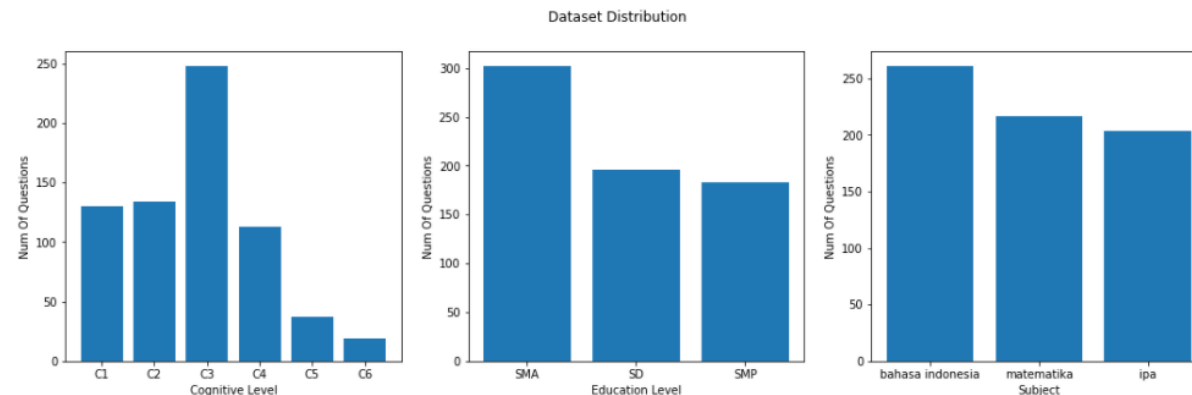


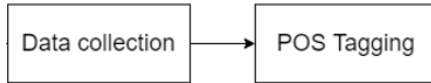
Data Collection

Sumber:

- EduBox (Pembimbing)
- Artikel Blog RuangGuru
- Penelitian Syarifah et al. [15]

	A	B	C	D	E	F
1	text	question	options	ans	label	category
2	"Apa ?" tanya Semut	Pada cuplikan fabel	A. 1 B. 2 C. 3 D. 4	B	C2	Mengkategorikan
3	Suatu hari Si Kancil,	Kata sandang yang te	B. di pada di pinggir	A	C2	Mengkategorikan
4	Perhatikan kutipan	Karakter tokoh Beru	B. Penyayang	A	C2	Memilih
5		Cerita rakyat pada za	B. hikayat	A	C1	Menyebutkan
6		Ciri bahasa dalam fal	A. naratif B. monolog	A	C1	Menyebutkan
7		Puisi rakyat atau pui	terpesona, terhina.	C	C2	Memilih
8		Surat pribadi adalah.	komunikasi tulis	A	C1	Menyebutkan
9		Surat di atas menunj	B. undangan	B	C2	Memilih
10		Ciri teks prosedur ad	objek dengan cara	B	C1	Menyebutkan
11		Struktur teks prosed	A. Tujuan--langkah-l	C	C1	Menyebutkan
12		Surat penawaran, Su	B. Dinas	C	C1	Memilih
13	Secara garis besar bu	Yang termasuk unsu	buku.	C	C1	Memilih
14		Rangkuman adalah h	ide pokok (gagasan	B	C1	Menyatakan
15	Pada zaman dahulu	Bagian Orientasi Fab	waktu, suasana,	B	C2	Menjelaskan
16	Saya kira tidak ada d	Cuplikan fabel terse		C	C2	Mengkategorikan
17	Kemudian Monyet n	'Cuplikan fabel terse		B	C2	Mengkategorikan
18	Monyet berkata, "Ke	Pernyataan tersebut		C	C2	Mengkategorikan
19	Pada suatu hari, Mor	Cuplikan fabel terse	B. Komplikasi	A	C2	Mengkategorikan
20	Sore hari yang cerah	Di pada kata di huta	A. kata seru	B	C2	Mengkategorikan
21	Sore hari yang cerah	Di pada kata diintai t	A. kata seru	D	C2	Mengkategorikan
22	Lina Malia Jalan Jing	Bagian surat tersebu	A. langsung	B	C2	Mengkategorikan
23	"Jangan! Jangan!" si	Kalimat tersebut ter		A	C2	Mengkategorikan
24	Burung Hantu selalu	Pada kutipan cerita c	A. kalimat pertama E	D	C2	Mengkategorikan



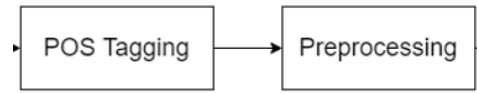


Part-of-Speech Tagging (POS Tagging)

- Menggunakan Pre-Trained model dari FlairNLP.
- Data disimpan pada DataFrame.

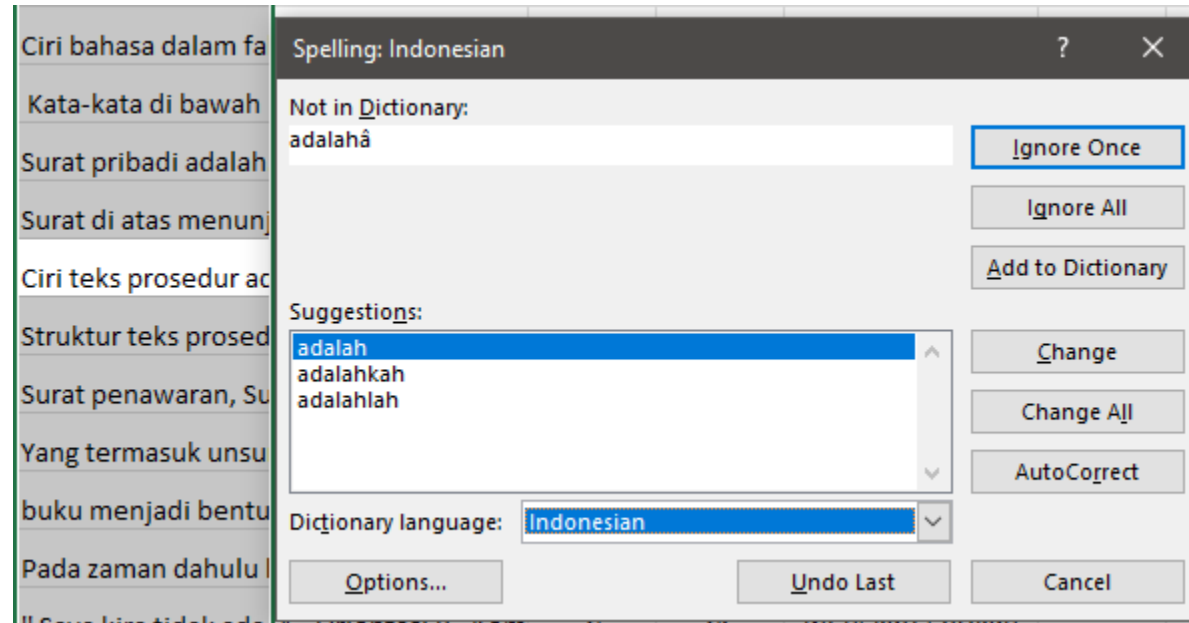
tag_pair
[[Makna, NOUN], [istilah, NOUN], [kata, NOUN], ...]
[[Latar, NOUN], [suasana, NOUN], [pada, ADP], ...]
[[Makna, NOUN], [frasa, NOUN], [cokelat, NOUN], ...]
[[Maksud, NOUN], [pernyataan, NOUN], [Evakuasi, ...]]
[[Nilai, NOUN], [moral, NOUN], [pada, ADP], [k...]]

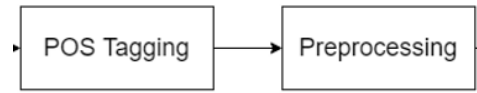




Preprocessing

Pemeriksaan ejaan





Preprocessing

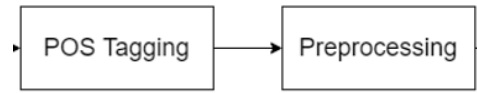
Casefolding

```
[['makna', 'NOUN'], ['istilah', 'NOUN'], ['kata', 'NOUN'], ['vulkanis', 'NOUN'], ['pada', 'ADP'], ['kutipan', 'NOUN'], ['teks', 'NOUN'], ['tersebut', 'DET'], ['adalah', 'AUX'], ['...', 'PUNCT'], ['.', 'PUNCT']]
```

Punctuation Removal

```
[['makna', 'NOUN'], ['istilah', 'NOUN'], ['kata', 'NOUN'], ['vulkanis', 'NOUN'], ['pada', 'ADP'], ['kutipan', 'NOUN'], ['teks', 'NOUN'], ['tersebut', 'DET'], ['adalah', 'AUX']]
```





Preprocessing

Stopwords Removal

Membuat 2 data berbeda untuk stopwords default PySastrawi dan modifikasi

```
sw_keep = ['adalah', 'apa', 'arti', 'artinya', 'berapa', 'berapakah', 'beri',  
           'berikan', 'diantaranya', 'disebut', 'jelaskan', 'karena',  
           'mengapa', 'menunjukkan', 'merupakan', 'rupa', 'sebut']
```

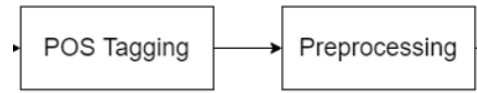
Default

```
In [22]: df['tag_pair_def'][0]  
Out[22]: [['makna', 'NOUN'],  
          ['istilah', 'NOUN'],  
          ['vulkanis', 'NOUN'],  
          ['kutipan', 'NOUN'],  
          ['teks', 'NOUN']]
```

Modifikasi

```
In [23]: df['tag_pair_mod'][0]  
Out[23]: [['makna', 'NOUN'],  
          ['istilah', 'NOUN'],  
          ['vulkanis', 'NOUN'],  
          ['kutipan', 'NOUN'],  
          ['teks', 'NOUN'],  
          ['adalah', 'AUX']]
```





Preprocessing

Stemming

Mengubah kata kedalam bentuk dasarnya

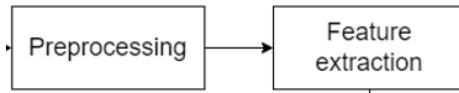
Sebelum

```
In [22]: df['tag_pair_def'][0]
Out[22]: [['makna', 'NOUN'],
           ['istilah', 'NOUN'],
           ['vulkanis', 'NOUN'],
           ['kutipan', 'NOUN'],
           ['teks', 'NOUN']]
```

Sesudah

```
In [26]: df.tag_pair_def = stem(df.tag_pair_def)
          df.tag_pair_def[0]
Out[26]: [['makna', 'NOUN'],
           ['istilah', 'NOUN'],
           ['vulkanis', 'NOUN'],
           ['kutip', 'NOUN'],
           ['teks', 'NOUN']]
```





Feature Extraction

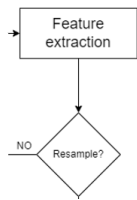
Contoh Nilai TF-IDF dan TFPOS-IDF untuk dokumen nomor 6 pada dataset

	Teladan (NOUN)	Tokoh (NOUN)	Dasar (ADP)	Kutip (NOUN)
TF-IDF	0.402	0.521	0.638	0.402
TFPOS-IDF	0.689	0.563	0.145	0.434

Tabel 1 Hasil Ekstraksi Fitur Pada Dokumen nomor 6

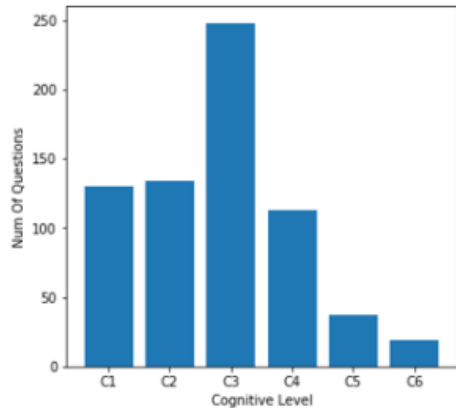
*nilai dibulatkan ke atas.





Random Oversampling

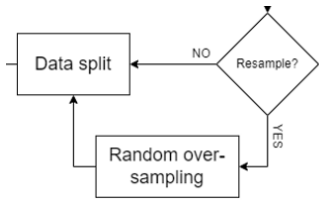
- Digunakan untuk mengatasi data yang tidak seimbang
- Implementasi menggunakan Imbalanced-learn, strategi 'not majority'



Kelas	Sebelum random oversampling	Setelah random oversampling
C1	130	248
C2	134	248
C3	248	248
C4	113	248
C5	37	248
C6	19	248

Tabel 2 Hasil Random Oversampling





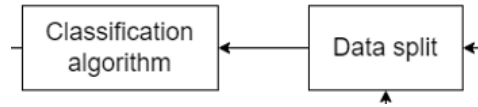
Data Split

- Data dipisah dengan rasio 80:20 untuk training dan testing
- Parameter random_state = 23 untuk pembagian data yang konsisten tiap eksekusi

Skenario	Algoritma	Feature Extraction	Stopwords	Random Over-Sampling
1	SVM	TF-IDF	Default	N
	NB	TF-IDF	Default	N
2	SVM	TF-IDF	Modifikasi	N
	NB	TF-IDF	Modifikasi	N
3	SVM	TF-IDF	Default	Y
	NB	TF-IDF	Default	Y
4	SVM	TF-IDF	Modifikasi	Y
	NB	TF-IDF	Modifikasi	Y
5	SVM	TFPOS-IDF	Default	N
	NB	TFPOS-IDF	Default	N
6	SVM	TFPOS-IDF	Modifikasi	N
	NB	TFPOS-IDF	Modifikasi	N
7	SVM	TFPOS-IDF	Default	Y
	NB	TFPOS-IDF	Default	Y
8	SVM	TFPOS-IDF	Modifikasi	Y
	NB	TFPOS-IDF	Modifikasi	Y

Tabel 3 Skenario Pengujian

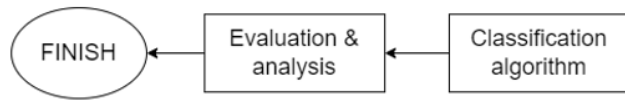




Classification Algorithm

- Support Vector Machine (SVM)
- Parameter $C = 1$ dan kernel = linear
- Naïve Bayes (NB)
- MultinomialNB(), Parameter *default*
- Parameter akan dioptimasi menggunakan GridSearchCV





Evaluation & analysis

$$Accuracy = \frac{TP + TN}{n} \quad (7)$$

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F1-Measure = \frac{2 \times (Precision + Recall)}{(Precision + Recall)} \quad (10)$$

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 6 Confusion Matrix [13]



4. Evaluasi

Analisis Hasil Pengujian

Skenario	Algoritma	Feature Extraction	Stopwords	Random Over-Sampling	Akurasi & F1-Measure	Akurasi & F1-Measure optimized
1	SVM	TF-IDF	Default	N	0.453 0.429	0.445 0.438
	NB	TF-IDF	Default	N	0.423 0.421	0.445 0.434
2	SVM	TF-IDF	Modifikasi	N	0.474 0.452	0.438 0.434
	NB	TF-IDF	Modifikasi	N	0.401 0.421	0.467 0.479
3	SVM	TF-IDF	Default	Y	0.799 0.798	0.839 0.837
	NB	TF-IDF	Default	Y	0.772 0.772	0.815 0.815

4	SVM	TF-IDF	Modifikasi	Y	0.819 0.818	0.842 0.842
	NB	TF-IDF	Modifikasi	Y	0.782 0.781	0.829 0.829
5	SVM	TFPOS-IDF	Default	N	0.438 0.43	0.438 0.43
	NB	TFPOS-IDF	Default	N	0.431 0.463	0.453 0.45
6	SVM	TFPOS-IDF	Modifikasi	N	0.445 0.431	0.504 0.491
	NB	TFPOS-IDF	Modifikasi	N	0.401 0.43	0.467 0.472
7	SVM	TFPOS-IDF	Default	Y	0.815 0.814	0.836 0.836
	NB	TFPOS-IDF	Default	Y	0.735 0.732	0.795 0.793
8	SVM	TFPOS-IDF	Modifikasi	Y	0.826 0.825	0.846 0.846
	NB	TFPOS-IDF	Modifikasi	Y	0.752 0.748	0.812 0.81

- Random Oversampling dapat meningkatkan performa klasifikasi pada semua scenario.
- Modifikasi stopwords & TFPOS-IDF secara rata-rata dapat meningkatkan performa klasifikasi.



4. Evaluasi

Analisis Hasil Pengujian

Algoritma	Feature Extraction	Stopwords	Random Over-Sampling	Akurasi & F1-Measure	Akurasi & F1-Measure optimized
SVM	TFPOS-IDF	Modifikasi	Y	0.826	0.846
				0.825	0.846
NB	TF-IDF	Modifikasi	Y	0.782	0.829
				0.781	0.829

Parameter SVM, C = 10 & kernel = linear

Parameter NB, alpha = 0.0



4. Evaluasi

Analisis Hasil Pengujian

	C1	C2	C3	C4	C5	C6
SVM	4	16	21	8	0	0
NB	5	14	21	9	0	3

Tabel 5 *Jumlah Salah Prediksi per Kelas*

	Bahasa Indonesia	IPA	Matematika
SVM	22	9	18
NB	26	9	19

Tabel 6 *Jumlah Salah Prediksi per Mata Pelajaran*



4. Evaluasi

Analisis Hasil Pengujian

	Predicted C4, Actual C3	Predicted C2, Actual C3
SVM	8	7
NB	7	9

Contoh soal misklasifikasi Predicted C2, Actual C3	Contoh Soal kelas C2
Berikut merupakan contoh kalimat untuk iklan buku tulis..	Berikut merupakan ciri ciri dari iklan baris , kecuali ..

Contoh soal misklasifikasi Predicted C4 Actual C3	Contoh Soal kelas C4
Ditentukan $\sin A = \frac{1}{2}$,maka nilai $\cos 2A$ adalah	Pada segitiga ABC diketahui panjang sisi $AB = 10$ cm, sisi $AC = 12$ cm, dan $\sin B = \frac{1}{2}$ maka nilai $\cos C$ adalah



5. Kesimpulan

- Pada penelitian ini, performansi SVM dapat mengungguli performansi NB.
- Ekstraksi fitur dengan TFPOS-IDF dapat memberikan performansi yang lebih baik dibandingkan TF-IDF pada algoritma SVM.
- TF-IDF memiliki performansi yang lebih baik dibandingkan dengan TFPOS-IDF pada algoritma NB
- Memodifikasi *stopwords* dapat membantu memberikan performansi yang baik untuk kedua algoritma.
- Melakukan random oversampling pada data dapat meningkatkan performa yang dihasilkan untuk algoritma SVM dan NB.



Terima Kasih

