Journal of Informatics Engineering and Software Applications

Personality Classification on Social Media Using Lexicon Construction with Decision Tree Algorithm

Deliana[1], Muhammad Arhami[2]*, Musta'inul Abdi[3], Umri Erdiansyah[4]

[1,2,3,4] Jurusan Tekniknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe Jln. B.Aceh Medan Km.280 Buketrata 24301 INDONESIA
*Penulis Korespondensi : muhammad.arhami@pnl.ac.id

INFORMASI ARTIKEL

Riwayat artikel:
Diajukan pada 12 Mei 25
Direvisi pada 29 Mei 25
Publikasi pada 20 Juni 25

Kata kunci:
Klasifikasi Kepribadian
Lexicon
Decision Tree
Myers-Briggs Type Indicator (MBTI)

Keywords:
Personality Classification,
Lexicon
Decision Tree
Myers-Briggs Type Indicator
(MBTI)

ABSTRAK

Perkembangan teknologi informasi dan komunikasi telah mendorong munculnya berbagai platform media sosial yang memungkinkan individu untuk berinteraksi dan mengekspresikan diri. Salah satu aspek penting yang dapat dianalisis dari data di media sosial adalah kepribadian pengguna. Penelitian ini bertujuan untuk mengklasifikasikan kepribadian pengguna media sosial dengan menggunakan pendekatan berbasis leksikon yang dioptimalkan melalui algoritma *Decision Tree*. Algoritma *Decision Tree* dipilih karena kemampuannya dalam mempartisi data menjadi subset yang lebih kecil berdasarkan fitur tertentu, sehingga menghasilkan model prediktif yang efektif. Penelitian ini menggunakan data dari platform Kaggle dan fokus pada analisis teks untuk mengidentifikasi kepribadian berdasarkan 16 tipe kepribadian yang ditentukan oleh Myers-Briggs Type Indicator (MBTI). Proses klasifikasi melibatkan beberapa tahap, termasuk prapemrosesan data, pembobotan TF-IDF, dan penerapan model *Decision Tree*. Hasil penelitian menunjukkan bahwa pendekatan berbasis leksikon yang diimplementasikan dengan algoritma *Decision Tree* dapat menghasilkan akurasi yang cukup tinggi dalam klasifikasi kepribadian pengguna media sosial. Penelitian ini memberikan kontribusi pada pengembangan model klasifikasi kepribadian yang dapat digunakan dalam berbagai aplikasi seperti pemasaran, perekrutan tenaga kerja, dan pengembangan produk. Namun, penelitian ini juga mengakui adanya keterbatasan dalam akurasi model yang dipengaruhi oleh variabilitas penggunaan bahasa di media sosial.

ABSTRACT

The development of information and communication technology has led to the emergence of various social media platforms that allow individuals to interact and express themselves. One important aspect that can be analyzed from data on social media is the user's personality. This research aims to classify the personality of social media users by using a lexicon-based approach optimized through the Decision Tree algorithm. The Decision Tree algorithm was chosen for its ability to partition data into smaller subsets based on specific features, resulting in an effective predictive model. This research uses data from the Kaggle platform and focuses on text analysis to identify personalities based on 16 personality types defined by the Myers-Briggs Type Indicator (MBTI). The classification process involves several stages, including data preprocessing, TF-IDF weighting, and application of the Decision Tree model. The results show that the lexicon-based approach implemented with the Decision Tree algorithm can produce high accuracy in social media user personality classification. This research contributes to the development of personality classification models that can be used in various applications such as marketing, workforce recruitment, and product development. However, this research also

recognizes the limitations in the accuracy of the model which is influenced by the variability of language use in social media.

1. Pendahuluan

Teknologi informasi telah membawa perubahan besar dalam cara manusia berinteraksi, terutama melalui media sosial. Salah satu aspek yang dapat dipelajari dari interaksi di media sosial adalah kepribadian pengguna. Kepribadian merupakan kata yang berasal dari bahasa latin, persona yang berarti topeng teatrikal yang digunakan oleh para *actor* [1]. Penelitian mengenai analisis kepribadian di media sosial telah menjadi topik yang menarik dalam bidang psikologi dan ilmu komputer. Kepribadian individu berperan penting dalam mempengaruhi pola interaksi, pengelolaan emosi, serta pengambilan keputusan. Dengan demikian, analisis kepribadian melalui jejak digital di media sosial dapat memberikan wawasan yang signifikan bagi berbagai aplikasi, termasuk pemasaran, rekrutmen tenaga kerja, serta pengembangan produk.

Myers Briggs Type Indicator (MBTI) adalah sistem tipe kepribadian yang membagi setiap orang ke dalam 16 tipe kepribadian yang berbeda di 4 sumbu yaitu ; *Introversion (I) – Extroversion (E), Intution (N) – Sensing (S), Thingking (T) – Feeling (F), Judging (J) – Perceiving (P)* [2]. Analisis faktor terhadap berbagai daftar kata sifat yang digunakan dalam kuesioner deskripsi kepribadian telah digunakan untuk mendapatkan empat ciri kepribadian ini berulang kali. Bahasa mengandung perbedaan yang paling signifikan antara orang-orang, dan semakin signifikan perbedaan tersebut, semakin besar kemungkinannya untuk diungkapkan dalam satu kata. Metode berbasis *lexicon* adalah salah satu metode yang efektif untuk menganalisis teks dan menemukan kepribadian berdasarkan kata-kata yang digunakan oleh pengguna media sosial. *Lexion* merupakan sumber daya *Lexical* yang ditingkatkan secara *Explicit* dirancang untuk mendukung klasifikasi sentimen dan aplikasi pengembangan opini [3].

Penelitian ini mencakup pengoptimalan konstruksi *Lexicon* menggunakan algoritma *Decision Tree* untuk mengatasi tantangan dalam identifikasi dan interpretasi kata atau frasa yang mencerminkan karakteristik kepribadian. Algoritma dari *Decision Tree* yang digunakan dalam penelitian ini adalah noktah keputusan yang selalu bercabang biner [4]. Algoritma *Decision Tree* adalah salah satu teknik *machine learning* yang dapat digunakan untuk klasifikasi data. Algoritma *Decision Tree* merupakan salah satu teknik *machine learning* yang digunakan dalam klasifikasi data. Algoritma ini berfungsi dengan membagi menjadi subset yang lebih kecil berdasarkan fitur-fitur tertentu, sehingga menghasilkan model berbentuk pohon yang dapat digunakan untuk prediksi.

Dalam konteks klasifikasi kepribadian di media sosial, *Decision Tree* berperan dalam mengelompokkan kepribadian pengguna ke dalam berbagai tipe kepribadian berdasarkan analisis leksikal dari tweet, caption, dan komentar yang dihasilkan oleh pengguna.

Terdapat sejumlah permasalahan terkait klasifikasi kepribadian pengguna media sosial dalam merespons isu-isu yang sedang *trending*, antara lain permasalahan dalam klasifikasi kepribadian pengguna sosial media terkait isu trending mencakup variabilitas penggunaan bahasa, seperti slang dan bahasa daerah, yang menyulitkan akurasi model klarifikasi, dan *postingan trending* sering penuh dengan emosi, sehingga analisis harus membedakan emosi dari karakteristik kepribadian yang stabil. Klasifikasi

kepribadian diharapkan mampu mengatasi berbagai permasalahan yang dihadapi oleh pengguna media sosial dalam merespons isu-isu yang sedang berkembang, sehingga dapat meningkatkan presisi dan relevansi dalam interaksi sosial di platform tersebut.

2. Metode

2.1 Teknik Pengumpulan Data

Data yang digunakan dalam penelitian ini bersumber dari platform Kaggle berujumlah 17.352 entri data yang mencakup teks yang diposting pengguna di media sosial. Metode yang digunakan adalah (NLP) dengan konstruksi Lexicon dan algoritma *Decision Tree*. *Natural Languange Processing* (NLP) adalah bagian dari *Artifical Intelligence* yang mengupayakan agar komputer dapat memahami dan memberikan *output* dalam bentuk bahasa manusia[4]. *Natural Languange* adalah bahasa yang digunakan menusia untuk berinteraksi [5]. Bahasa manusia yang kompleks menjadi tantangan tersendiri bagi computer untuk dapat mengerti dan memahami bahasa manusia, yaitu *Syntatic Analysis* dan *Semantic Analysis* merujuk pada makna kata, frasa, maupun kalimat dalam sebuah Bahasa.

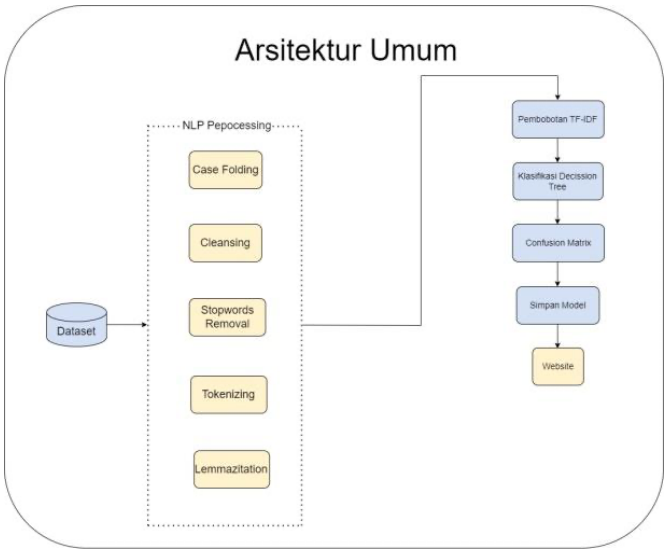
Dalam proses *text preprocessing*, terdapat beberapa tahapan penting yang dilakukan untuk mempersiapkan data teks agar lebih terstruktur dan siap diproses lebih lanjut. Salah satu tahapan awal yang paling sederhana namun sering diabaikan adalah *Case Folding*. Proses ini bertujuan untuk mengubah seluruh huruf dalam teks menjadi huruf kecil, sehingga data menjadi lebih konsisten dan tidak terjadi duplikasi makna hanya karena perbedaan kapitalisasi huruf. Selanjutnya, terdapat proses *Cleaning* yang berfungsi untuk membersihkan teks dari berbagai tanda baca seperti \$, ?, #, @, dan simbol lainnya yang tidak dibutuhkan dalam analisis teks. Tahapan ini penting untuk menyederhanakan data dan menghilangkan elemen-elemen yang tidak relevan.

Setelah data bersih, tahap berikutnya adalah *Tokenizing*. *Tokenizing* merupakan proses memecah teks atau kalimat menjadi bagian-bagian yang lebih kecil, yaitu kata per kata. Dengan proses ini, setiap kata dapat dianalisis secara terpisah, sehingga memudahkan dalam membedakan antara kata yang bermakna dan kata yang tidak bermakna atau hanya berfungsi sebagai kata hubung. Dalam bahasa pemrograman seperti Python, proses ini juga sering disertai dengan penghapusan angka, tanda baca, dan spasi yang berlebihan. Selanjutnya, dilakukan proses *Stopwords Removal*, yaitu penghapusan kata-kata umum yang dianggap tidak memiliki makna penting dalam analisis, seperti "*when*", "*or*", "*this*", "*at*", dan sebagainya. Kata-kata ini dianggap sebagai *noise* dalam data teks dan biasanya dihapus untuk meningkatkan efisiensi analisis. Misalnya, kalimat "Angelina Jolie its the best actor woman for ever" setelah diproses akan menjadi "Angelina Jolie best woman actor".

Tahapan terakhir dalam *text preprocessing* adalah *Lemmatization*. *Lemmatization* bertujuan untuk mengembalikan setiap kata ke bentuk dasarnya (*lemma*) dengan cara menghapus awalan atau imbuhan yang menyebabkan perubahan bentuk kata. Sebagai contoh, kata "*underdevelopment*" akan dikembalikan ke bentuk dasarnya yaitu "*develop*". Proses ini penting untuk menyatukan berbagai variasi kata menjadi satu bentuk dasar yang sama, sehingga analisis terhadap makna kata menjadi lebih akurat dan menyeluruh. Seluruh tahapan ini saling berkaitan dan sangat penting dalam menghasilkan data teks yang bersih, konsisten, dan siap untuk dianalisis lebih lanjut dalam proses *Natural Language Processing* (NLP).

2.2 Aristektur Umum

Perancangan sistem klasifikasi kepribadian terdiri dari sejumlah tahapan yang dilakukan secara sistematis, seperti yang ditunjukkan pada Gambar 1. Tahapan-tahapan ini meliputi berbagai langkah penting yang dimulai dari pengumpulan data, prapemrosesan data, pemilihan fitur, hingga penerapan algoritma klasifikasi. Setiap tahap dirancang untuk memastikan bahwa sistem dapat bekerja secara optimal dalam mengklasifikasikan kepribadian berdasarkan data yang tersedia, dengan tujuan menghasilkan model yang akurat dan dapat diandalkan dalam analisis lebih lanjut.



Gambar 1. Arsitektur Umum

Gambar 1. Arsitektur Umum menunjukkan desain umum dari perancangan *machine learning*. Sebelum memulai tahapan berikutnya, *dataset* diperlukan untuk proses *training* dan *testing*. Berdasarkan sumber data yang diperoleh berjumlah 17352 data.

2.3 Dataset

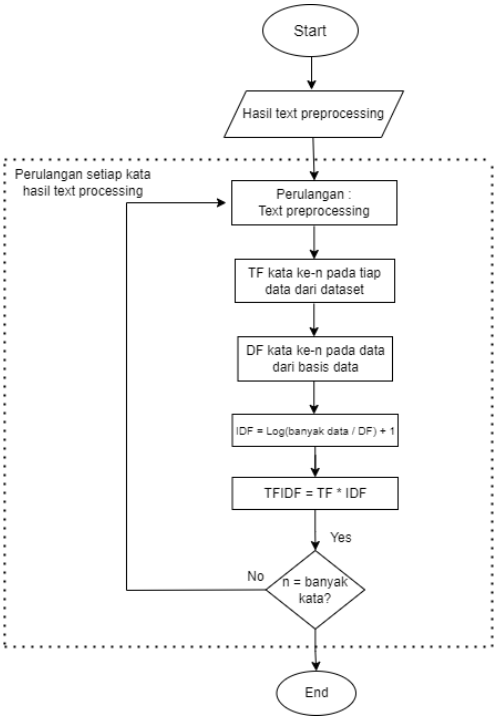
Dataset yang digunakan dalam penelitian ini diperoleh secara eksklusif dari situs web resmi Kaggle https://www.kaggle.com/datasets/datasnaek/mbti-type berjumlah 17352 data. Setelah melalui proses pengujian data terserbut, pada penelitian ini diambil sebanyak 52056 data.

2.4 *Preprocessing Data*

Preprocessing dilakukan untuk membersihkan data yang belum terstruktur, yang kemudian dapat digunakan untuk membuat model *machine learning.*

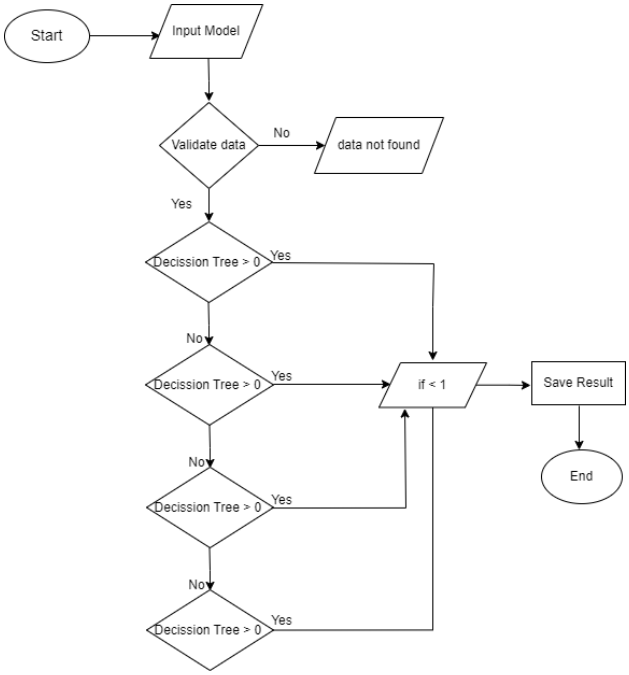
2.5 Pembobotan TF-IDF

Pembobotan TF-IDF dilakukan untuk menentukan tingkat frekuensi kemunculan suatu kata secara relatif dalam dokumen. Alur kerja metode ini diilustrasikan pada Gambar 2. Setelah melalui tahap praproses, diperoleh *dataset* yang telah ditokenisasi pada setiap data atau kalimat. Selanjutnya, frekuensi kemunculan kata dalam setiap data dihitung. Berdasarkan proses ini, diperoleh nilai *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF). Kedua nilai tersebut kemudian dikalikan untuk memperoleh bobot TF-IDF.



Gambar 2. Flowchart Pembobotan TF-IDF

2.6 Klasifikasi Kepribadian menggunakan metode *Decision Tree*



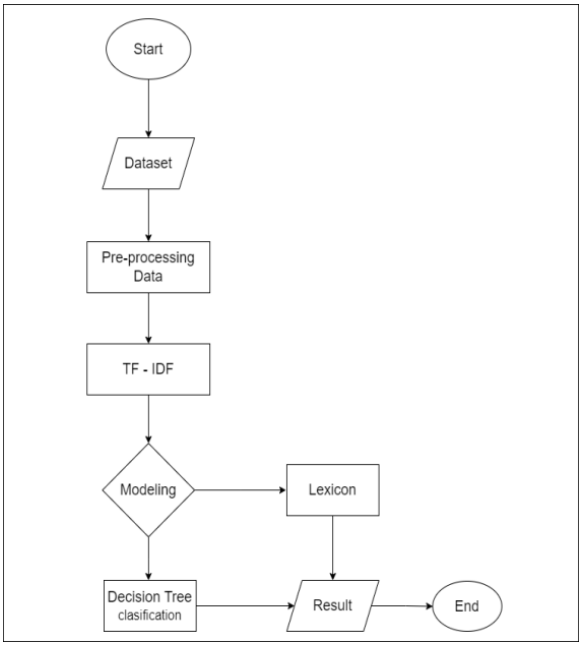
Gambar 3. Flowchart Klasifikasi Kepribadian Menggunakan Metode *Decision Tree*

Decision Tree merupakan algoritma *machine learning* yang digunakan untuk mengoptimalkan model klasifikasi dengan membagi *dataset* berdasarkan fitur yang paling informatif pada setiap *node*. Proses ini berlanjut secara rekursif hingga tercapai kondisi penghentian, sehingga menghasilkan pohon keputusan yang dapat digunakan untuk regresi atau klasifikasi. Pada penelitian ini, melatih suatu *model Decision Tree* dengan beberapa *class* sekaligus, *Decision Tree* akan mencari setiap *node* yang dapat memisahkan keempat *class* di ruang fitur.

2.7 Evaluasi Model Menggunakan Metde *Confusion Matrix*

Dalam analisis klasifikasi, *Confusion Matrix* merupakan alat evaluasi yang sangat penting untuk menilai kinerja sebuah model atau sistem klasifikasi. Metode ini bekerja dengan cara membandingkan hasil prediksi model terhadap data uji dengan nilai sebenarnya, sehingga memberikan gambaran sejauh mana akurasi dan kesalahan yang dilakukan oleh model tersebut. *Confusion Matrix* terdiri dari empat komponen utama yang mengklasifikasikan hasil prediksi ke dalam beberapa kategori. Pertama, *True Positive* (TP), yaitu jumlah kasus di mana model secara akurat memprediksi hasil positif ketika hasil sebenarnya juga positif. Kedua, *True Negative* (TN), yaitu jumlah kasus di mana model dengan benar memprediksi hasil negatif ketika hasil sebenarnya juga negatif. Ketiga, *False Positive* (FP), yaitu jumlah kasus di mana model salah memprediksi hasil positif, padahal kenyataannya hasil tersebut negatif. Terakhir, *False Negative* (FN), yaitu jumlah kasus di mana model salah memprediksi hasil negatif, padahal sebenarnya hasil tersebut positif. Keempat komponen ini sangat penting dalam mengevaluasi performa model klasifikasi, khususnya dalam menghitung metrik seperti akurasi, presisi, *recall*, dan F1-score.

2.8 Penerapan NLP dengan Lexicon dan *Decision Tree*



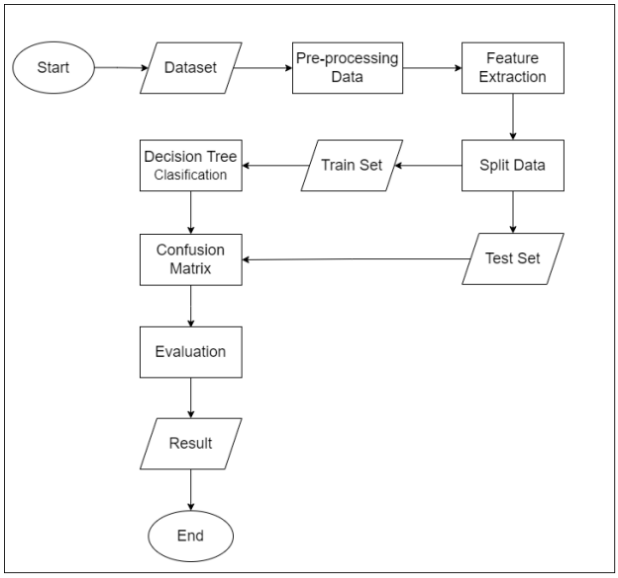
Gambar 4. Flowchart Penerapan NLP dengan Konstruksi *Lexicon* dan *Decison Tree*

Pada Gambar 4. Flowchart Penerapan NLP dengan Konstruksi Lexicon dan *Decision Tree*, dijelaskan tahapan proses alur sistem. Proses dimulai dengan memasukkan dataset dalam format .csv, diikuti dengan tahap praproses data yang mencakup proses pembersihan (*cleansing*), penokenan (*tokenizing*), normalisasi huruf (*casefolding*), penghapusan kata tidak bermakna (*stopword removal*), dan *lemmatization*. Setelah tahap praproses selesai, dilakukan proses TF-IDF untuk mengubah teks menjadi representasi numerik. Selanjutnya, tahap pemodelan dibagi menjadi dua proses, yaitu penggunaan *Lexicon* dan klasifikasi menggunakan algoritma *Decision Tree*, yang hasilnya akan ditampilkan dalam bagian hasil (*results*).

2.9 Perancangan Klasifikasi *Decision Tree*

Gambar 5 menjelaskan tahapan proses klasifikasi menggunakan metode *Decision Tree*. Tahapan dimulai dengan menginput dataset yang kemudian dilanjutkan dengan proses prapemrosesan data, yang

mencakup *cleansing*, *tokenizing*, *case folding*, *stopword removal*, dan *lemmatization*. Selanjutnya, dilakukan ekstraksi fitur dari hasil prapemprosesan tersebut untuk mendapatkan informasi yang relevan. Dalam analisis klasifikasi, *Confusion Matrix* merupakan alat evaluasi yang sangat penting untuk menilai kinerja sebuah model atau sistem klasifikasi.



Gambar 5. Flowchart Perancangan Klasifikasi *Decision Tree*

Tahap berikutnya adalah pembagian melalui proses *split data*, di mana dibagi menjadi dua bagian, yaitu 80% sebagai data uji dan 20% sebagai data latih. Setelah itu, klasifikasi dilakukan menggunakan algoritma *Decision Tree*. Proses ini diikuti oleh pengujian akurasi model menggunakan *Confusion Matrix*. Tahap terakhir adalah evaluasi model, di mana klasifikasi dan pengujian ditampilkan..

2.10 Rancangan Analisis Pengujian *Confusion Matrix*

Data yang yang telah melalui tahap prapemrosesan akan dibagi menjadi dua kategori, yaitu data uji dan data latih. Skenario pengujian dilakukan untuk mengamati pembagian jumlah data, sebanyak 53.056 data akan dipisahkan ke dalam dua kategori tersebut. Rincian pembagian data ini disajikan pada Tabel 1.

Tabel 1. Jumlah pembagian data latih dan data uji

Data latih	Data uji
80% data	20% data

Confusion Matrix merupakan salah satu metode yang digunakan untuk mengukur kinerja metode klasifikasi.

2.10.1 *Recall*

Recall adalah jumlah data dengan kategori positif yang diklasifikasikan dengan benar oleh sistem dibandingkan dengan semua data positif yang ada[7].

$$Recal = \frac{TP}{FN+TP} \quad (1)$$

2.10.2 *Precision*

Presisi menunjukkan jumlah data dengan kategori positif yang diklasifikasikan dengan benar oleh sistem dibandingkan dengan semua data prediksi positif [7].

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

2.10.3 *Accuracy*

Akurasi menunjukkan jumlah data yang diklasifikasikan dengan benar dibandingkan dengan total data [7].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

2.10.4 F1-Score

F1-Score menunjukkan rata-rata harmonis dari *precission* dan *recall* [7].

$$F1-Score = \frac{2\ x\ precision\ x\ recall}{precision+recall} \quad (4)$$

Confusion Matrix adalah sebuah formula untuk menganalisis pengklasifikasi dan seberapa baik pengklasifikasi tersebut mengenali tupel dari berbagai kelas. Terdapat empat istilah untuk mengukur performa menggunakan *Confusion Matrix*, yaitu, True Positive (TN) Data negatif yang terdeteksi dengan benar, False Positive (FP) Data negatif namun terdeteksi sebagai data positif, True Positive (TP) Data positif terdeteksi benar, False Negative (FN) Data positif terdeteksi sebagai data negatif.

3. Hasil Dan Pembahasan

Hasil penelitian ini akan menguraikan prosedur dan tahapan yang dilaksanakan sesuai dengan perancangan *model machine learning* yang telah dijelaskan sebelumnya.

3.1 Implementasi Rancangan

Bagian ini memberikan penjelasan menyeluruh tentang proses pengolahan bahasa natural, yang digunakan sebagai teknik untuk mempersiapkan dan membersihkan data teks mentah agar siap untuk analisis. Berikut adalah implementasi dari setiap prosedur.

3.1.1 *Cleansing Data*

	posts	clean
0	'http://www.youtube.com/watch?v=qsXHcwe3krw|||...	and intj moments sportscenter not top ten...
1	'I'm finding the lack of me in these posts ver...	i m finding the lack of me in these posts ver...
2	'Good one _____ https://www.youtube.com/wat...	good one course to which i say i k...
3	'Dear INTP, I enjoyed our conversation the o...	dear intp i enjoyed our conversation the o...
4	'You're fired.|||That's another silly misconce...	you re fired that s another silly misconce...
...	...	...
8670	'https://www.youtube.com/watch?v=t8edHB_h908||...	just because i always think of cats as fi do...
8671	'So...if this thread already exists someplace ...	so if this thread already exists someplace ...
8672	'So many questions when i do these things. I ...	so many questions when i do these things i ...
8673	'I am very conflicted right now when it comes ...	i am very conflicted right now when it comes ...
8674	'It has been too long since I have been on per...	it has been too long since i have been on per...

8675 rows × 2 columns

Gambar 6. Hasil *Cleansing Data*

Pada Gambar 6. Hasil *Cleansing Data* merupakan hasil proses *cleansing* yang sudah dilakukan pada kolom "*posts*" dan menampilkan hasilnya pada kolom "*clean*".

3.1.2 *Tokenization*

clean	tokenizing
and intj moments sportscenter not top ten...	(and, intj, moments, sportscenter, not, top, t...
i m finding the lack of me in these posts ver...	(i, m, finding, the, lack, of, me, in, these, ...
good one course to which i say i k...	(good, one, course, to, which, i, say, i, know...
dear intp i enjoyed our conversation the o...	(dear, intp, i, enjoyed, our, conversation, th...
you re fired that s another silly misconce...	(you, re, fired, that, s, another, silly, misc...
...	...
just because i always think of cats as fi do...	(just, because, i, always, think, of, cats, as...
so if this thread already exists someplace ...	(so, if, this, thread, already, exists, somepl...
so many questions when i do these things i ...	(so, many, questions, when, i, do, these, thin...
i am very conflicted right now when it comes ...	(i, am, very, conflicted, right, now, when, it...
it has been too long since i have been on per...	(it, has, been, too, long, since, i, have, bee...

Gambar 7. Hasil *Tokenizing*

Gambar 7. Hasil Tokenizing menunjukkan hasil dari proses tokenisasi yang telah diterapkan pada kolom *clean*. Proses ini memecah teks menjadi unit-unit kata yang lebih kecil, yang dikenal sebagai token. Selanjutnya, kolom tokenizing menampilkan hasil tokenisasi tersebut, di mana setiap kata dalam teks telah diidentifikasi sebagai token yang terpisah, yang nantinya akan digunakan dalam tahap analisis data selanjutnya.

3.1.3 *Lemmazitation* dan *Stopwordon*

Lemmatizing	type
and intj moment sportscenter not top ten play ...	INFJ
i m find the lack of me in these post very ala...	ENTP
good one course to which i say i know that s m...	INTP
dear intp i enjoy our conversation the other d...	INTJ
you re fire that s another silly misconception...	ENTJ

Gambar 8. Hasil *Lemmazitation* dan *Stopword*

Pada gambar 8. Hasil *Lemmazitation* dan *Stopword* merupakan hasil *lemma* dan *stopword* data yang sudah dilakukan proses *cleaning*, *token*, *stopword* dan menampilkan hasilnya pada kolom *lemmazitation*.

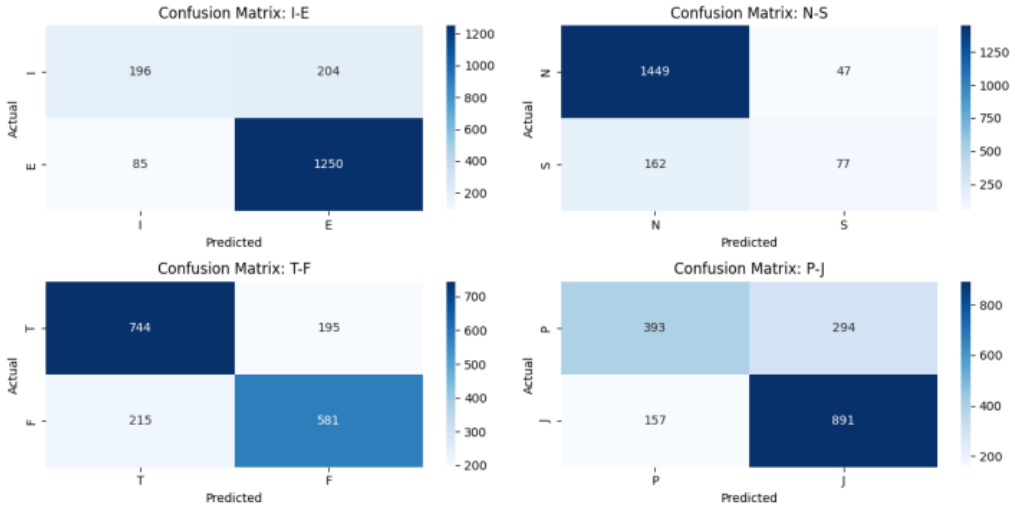
3.2 Implementasi Data menggunakan *Lexicon*

Berdasarkan Gambar 4.17, menjelaskan empat matrix untuk empat pasangan kepribadian: I-E, N-S, T-F, dan P-J. Performa model Algoritma *Decision Tree* untuk analisis kepribadian sebagai berikut:

```
type : ENFP
lexicon : ('E', 'N', 'F', 'P')
{'E': 0.5015906680805938, 'I': 0.4984093319194062, 'result': 'E'}
{'N': 0.5026567481402763, 'S': 0.4973432518597237, 'result': 'N'}
{'F': 0.5031914893617021, 'T': 0.4968085106382979, 'result': 'F'}
{'J': 0.4978768577494692, 'P': 0.5021231422505308, 'result': 'P'}

type : ENFP
lexicon : ('E', 'N', 'F', 'P')
{'E': 0.5019669551534225, 'I': 0.4980330448465775, 'result': 'E'}
{'N': 0.5031545741324921, 'S': 0.4968454258675079, 'result': 'N'}
{'F': 0.5019669551534225, 'T': 0.4980330448465775, 'result': 'F'}
{'J': 0.49724192277383766, 'P': 0.5027580772261623, 'result': 'P'}

type : INFJ
lexicon : ('I', 'N', 'F', 'J')
{'E': 0.4984051036682616, 'I': 0.5015948963317385, 'result': 'I'}
{'N': 0.5019952114924182, 'S': 0.4980047885075818, 'result': 'N'}
{'F': 0.501195219123506, 'T': 0.49880478087649405, 'result': 'F'}
{'J': 0.501195219123506, 'P': 0.49880478087649405, 'result': 'J'}

type : INFJ
lexicon : ('I', 'N', 'F', 'J')
{'E': 0.496996996996997, 'I': 0.503003003003003, 'result': 'I'}
{'N': 0.5052790346907994, 'S': 0.4947209653092006, 'result': 'N'}
...
{'N': 0.5059790732436472, 'S': 0.4940209267563528, 'result': 'N'}
{'F': 0.4981467753891772, 'T': 0.5018532246108228, 'result': 'T'}
{'J': 0.4981467753891772, 'P': 0.5018532246108228, 'result': 'P'}
```

Gambar 9. Hasil Implementasi Metode Lexicon

3.3 Hasil Pengujian *Confusion Matrix*



Gambar 9. Evaluasi Model Menggunakan *Confusion Matrix*

Gambar 9. Hasil Implementasi Metode *Lexicon* menjelaskan hasil dari implementasi metode *lexicon* dalam proses mencocokkan kata bedasarkan kamus dan diberi lebel variable kepribadian.

Tabel 2. Confusion Matrix untuk Kategori I-E (Introversion-Extroversion)

Actual \ Predicted	Introversion (I)	Extroversion (E)
Introversion (I)	196	204
Extroversion (E)	85	1250

Tabel 3. Confusion Matrix untuk Kategori N-S (Intuition-Sensing)

Actual \ Predicted	Intuition (N)	Sensing (S)
Intuition (N)	1449	47
Sensing (S)	162	77

Tabel 4. Confusion Matrix untuk Kategori T-F (Thinking-Feeling)

Actual \ Predicted	Thinking (T)	Feeling (F)
Thinking (T)	744	195
Feeling (F)	215	581

Tabel 5. Confusion Matrix untuk Kategori P-J (Perceiving-Judging)

Actual \ Predicted	Perceiving (P)	Judging (J)
Perceiving (P)	393	294
Judging (J)	157	891

4. Kesimpulan

Setelah melakukan perancangan dan pengujian klasifikasi kepribadian menunjukkan bahwa Algoritma *Decision Tree* dengan pendekatan berbasis leksikon efektif dalam mengklasifikasikan kepribadian pengguna media sosial, dengan tingkat akurasi mencapai 79,00% dalam mengidentifikasi berbagai tipe kepribadian berdasarkan analisis teks. Namun, meskipun model ini terbukti efektif, masih terdapat tantangan terkait variabilitas dan keragaman bahasa yang digunakan di media sosial, sehingga diperlukan penelitian lebih lanjut untuk meningkatkan kinerja model dalam menangani kompleksitas bahasa dan ekspresi pengguna yang beragam.

Referensi

[1] T. L. C. Yoong, N. R. Ngatirin, and Z. Zainol, "Personality prediction based on social media using *decision tree* algorithm," Pertanika J. Sci. Technol., vol. 25, no. S4, pp. 237–248, 2017.

[2] J. A. Sugihdharma and F. A. Bachtiar, "Myers-Briggs Type Indicator Personality Model Classification in English Text using Convolutional Neural Network Method," vol. 2, pp. 93–103, 2022..

[3] B. Liu, Web Data Mining. 2011.

[4] M. Hatta, "Stemmer Bahasa Indonesia Dengan Pendekatan Aturan," vol. 2, no. 7, pp. 1–11, 2022.

[5] R. M. Yanti, I. Santoso, and L. H. Suadaa, "Application of Named Entity Recognition via Twitter on SpaCy in Indonesian (Case Study : Power Failure in the Special Region of Yogyakarta)," vol. 4, no. 1, pp. 76–86, 2021.

[6] DQlab, "Begini Cara Implementasi Teknik Analisis Data untuk Text Preprocessing." https://dqlab.id/begini-cara-implementasi-teknik-analisis-data-untuk-text-preprocessing

[7] M. Liang, Data Mining: Concepts, Models, Methods, and Algorithms, vol. 36, no. 5. 2004..

[8] A. Souri, S. Hosseinpour, and A. M. Rahmani, "Personality classification based on profiles of social networks' users and the five-factor model of personality," Human-centric Comput. Inf. Sci., vol. 8, no. 1, 2018, doi: 10.1186/s13673-018-0147-4.

[9] M. H. Amirhosseini and H. Kazemian, "Machine Learning Approach to Personality Type Prediction Based on the Myers – Briggs Type Indicator ®," 2020.

[10] J. Media and I. Budidarma, "Feature Expansion Using Word2vec for Hate Speech Detection on Indonesian Twitter with Classification Using SVM and Random," vol. 6, no. April, pp. 979–988, 2022, doi: 10.30865/mib.v6i2.3855.

[11] S. Robertson and S. Robertson, "Understanding inverse document frequency : on theoretical arguments for IDF," 2006, doi: 10.1108/00220410410560582.