Perbandingan Metode Klasifikasi Support Vector Machine dan Stochastic Gradient Descent untuk Klasifikasi Teks Misogini di Reddit

1st Dhialif Fajarrahman
Informatika
Telkom University
Bandung, Indonesia
dhialif@student.telkomuniversity.ac.id

2nd Dade Nurjanah
Informatika
Telkom University
Bandung, Indonesia
dadenurjanah@telkomuniversity.ac.id

3rd Hani Nurrahmi
Informatika
Telkom University
Bandung, Indonesia
haninurrahmii@telkomuniversity.ac.id

***Abstrak* — Media sosial telah mempermudah orang-orang untuk menyampaikan opini dan kepercayaan mereka, namun kebebasan tersebut seringkali disalahgunakan untuk melakukan Online Abuse, terutama pelecehan pada perempuan atau disebut juga misogini. Penelitian ini dilakukan untuk memilih metode klasifikasi terbaik antara metode *Support Vector Machine* dan *Stochastic Gradient Descent* dengan menggunakan model klasifikasi Support Vector Classification dan *Stochastic Gradient Descent Classifier* untuk mengklasifikasi data misogini. Klasifikasi dilakukan terhadap dataset berisi teks yang mengandung unsur misogini dari media sosial Reddit. Metrik evaluasi yang dipertimbangkan berupa *Accuracy*, *Precision*, *Recall*, dan F1-Score yang dihitung melalui hasil yang didapat dari *Confusion Matrix*. Hasil yang didapatkan menunjukkan SVC dengan teknik Undersampling menghasilkan performa terbaik dalam mengidentifikasi konten misogini dengan *Accuracy* 82,84%, *Precision* 29%, *Recall* 71%, dan *F1-Score* 41%.**

Kata kunci*— deteksi misogini, klasifikasi teks, *support vector machine*, *stochastic gradient descent

I. PENDAHULUAN

Media sosial merupakan teknologi yang sudah sering digunakan. Di Indonesia sendiri, pengguna media sosial terdata sebesar 191,4 juta pengguna yaitu sekitar 68,9% total populasi Indonesia pada tahun 2022 [1]. Menurut Cambridge Dictionary [2], media sosial itu sendiri merupakan situs web atau program komputer yang memberikan orang-orang kemampuan untuk berkomunikasi dan berbagi informasi di internet melalui komputer atau ponsel. Selain sebagai kelebihan, kemampuan media sosial tersebut juga bisa menjadi kekurangan ketika digunakan oleh orang-orang yang tidak bertanggungjawab.

Online Abuse atau penyalahgunaan daring yaitu merupakan salah satu hal yang dapat terjadi yang diakibatkan penyalahgunaan media sosial. Menurut Get Safe Online [3], *Online Abuse* dapat berupa *cyberbullying*, *cyberstalking*, *trolling*, *creeping*, *doxxing*, dan sebagainya. Salah satu tipe *online abuse* yang sering terjadi yaitu *cyberbullying*, terutama pelecehan pada perempuan atau disebut juga sebagai misogini.

Misogini merupakan kebencian atau rasa tidak suka terhadap wanita yang diwujudkan dalam berbagai bentuk, termasuk diskriminasi seksual, fitnah perempuan, kekerasan terhadap perempuan, dan objektifikasi seksual perempuan [4]. Berdasarkan kasus yang dicatat pada Catatan Tahunan Komnas Perempuan Tahun 2020 [5], dari 2.389 kasus Kekerasan terhadap Perempuan yang dilaporkan ke Unit Pelayanan dan Rujukan Komnas Perempuan, dicatat bahwa 2.134 kasus merupakan kasus berbasis gender. Selain itu, tercatat bahwa Kekerasan Berbasis Gender Siber meningkat dari 126 kasus di 2019 menjadi 510 kasus di 2020.

Salah satu metode yang dapat dilakukan untuk mengatasi misogini di media sosial yaitu dengan mengembangkan sistem untuk mendeteksi misogini dengan melakukan klasifikasi teks. Klasifikasi teks dilakukan dengan mengidentifikasi teks yang berisi konten misogini dari media sosial dan mengklasifikasikannya menjadi data misoginis atau non-misoginis. Beberapa penelitian sudah pernah dilakukan untuk mengidentifikasi dan mengklasifikasi misogini dengan menggunakan berbagai metode klasifikasi teks yang berbeda seperti Long *Short-Term Memory* (LSTM)[4], *Bidirectional Encoder Representations From Transformers* (BERT)[4][6][7], *Generative Pre-trained Transformer* [7], *Support Vector Machine* (SVM)[8], *Random Forest* (RF)[8], dan *Gradient Boosted Trees* (GBT)[8].

Penelitian ini bertujuan untuk membandingkan performa dua metode klasifikasi, yaitu *Support Vector Machine* (SVM) dan *Stochastic Gradient Descent* (SGD), dalam mendeteksi misogini di Reddit dengan menggunakan dataset yang sudah tersedia. Dengan melakukan analisis komparatif ini, penelitian ini bertujuan untuk berkontribusi dalam mengembangkan solusi yang lebih efisien dan akurat untuk deteksi misogini di media sosial.

II. KAJIAN TEORI

Angeline dkk. [4] melakukan penelitian pendeteksian ucapan misogini menggunakan *Long Short-Term Memory*

(LSTM) dan *Bidirectional Encoder Representations From Transformers* (BERT) *Embeddings*. Penelitian dilakukan dengan membandingkan hasil performanya dengan *Logistic Regression* (LR) dan *Convolutional Neural Network* (CNN). Hasilnya yaitu LSTM dengan BERT *Embedding* memiliki nilai terbaik yaitu *Accuracy* 86,15%, *Precision* 84%, *Recall* 80%, dan *F1-Score* 81%. Penelitian ini menekankan pentingnya pemahaman kontekstual dalam deteksi misogini, ditunjukkan dengan keunggulan BERT untuk menangkap representasi kata yang memiliki nuance sehingga dapat memahami konteks dari data yang digunakan.

Untuk membantu penelitian untuk mendeteksi misogini, Ella Guest dkk. [9] menyajikan taksonomi hirarkis untuk misogini online dan dataset berlabel ahli dari 6567 postingan dan komentar di Reddit. Pada penelitian ini, peneliti melakukan percobaan klasifikasi menggunakan metode klasifikasi regresi logistik dan model BERT (baik yang berbobot maupun tidak berbobot) untuk mendapatkan performa baseline untuk mengklasifikasi biner konten misoginis. Model regresi logistik menghasilkan *Accuracy* 92%, *Precision* 88%, *Recall* 7%, dan *F1-Score* 13%. Model BERT tidak berbobot menghasilkan *Accuracy* 93%, *Precision* 67%, *Recall* 30%, dan *F1-Score* 42%. Sedangkan model BERT berbobot menghasilkan Accuracy 89%, *Precision* 38%, *Recall* 50%, dan *F1-Score* 43%. Ukuran dataset yang kecil dan ketidakseimbangan kelas ditekankan sebagai penyebab kesalahan klasifikasi.

Pada penelitiannya, Umar dkk. [10] menggunakan metode *Support Vector Machine* (SVM), *Random Forest* (RF), dan *Stochastic Gradient Descent* (SGD) untuk mengklasifikasi teks berisi kinerja programmer pada aktivitas media sosial. Hasil penelitiannya yaitu klasifikasi kinerja programmer dengan menggunakan k=10 *cross validation* menghasilkan SVM dengan *Accuracy* 81,3%, RF dengan *Accuracy* 74,4%, dan SGD dengan *Accuracy* 80,1%.

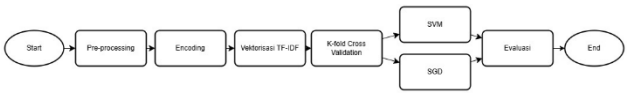
Pada penelitiannya, Oktanisa dan Supianto [11] melakukan perbandingan 9 teknik klasifikasi pada dataset Bank Direct Marketing. Metode yang digunakan yaitu *Support Vector Machine*, AdaBoost, Naïve Bayes, Constant, KNN, Tree, *Random Forest*, *Stochastic Gradient Descent*, dan CN2 Rule. Hasil yang didapatkan yaitu metode SGD memiliki hasil terbaik yaitu *Accuracy* 97,2%, *Precision* 94,5%, dan *Recall* 97,2%.

Pada penelitiannya, Prasetijo dkk. [12] menggunakan SVM dan SGD untuk membuat *Hoax Detection System* yang berbasis klasifikasi teks pada situs berita Indonesia. Model SVM yang digunakan yaitu SVM dengan kernel *linear*, sedangkan model SGD yang digunakan yaitu SGD dengan kernel *linear regression* dan SGD dengan *modifier-huber* Hasil yang didapatkan yaitu *Accuracy* tertinggi sebesar 86% dari model SGDClassifier yang menggunakan kernel *modifier-huber* terhadap situs 100 hoax dan 100 non-hoax.

III. METODE

A. Tahapan Penelitian

Penelitian ini dilakukan untuk mengetahui perbedaan kinerja antara model klasifikasi SVM dan SGD dalam mengklasifikasi dataset berisi konten misogini. Proses yang dilakukan yaitu termasuk pra-pemrosesan untuk mempersiapkan data, penggunaan *encoding* dan vektorisasi TF-IDF untuk transformasi data, dan lalu dilakukan pemisahan data menggunakan *K-Fold Cross Validation*. Selanjutnya, klasifikasi dilakukan menggunakan kedua model klasifikasi dengan hasilnya berupa *Accuracy*, *Precision*, *Recall*, dan *F1-Score*. Proses pengujian model dilakukan menggunakan Google Colab. Alur tahapan penelitian dapat dilihat pada gambar 1.



GAMBAR 1
(Tahapan Penelitian)

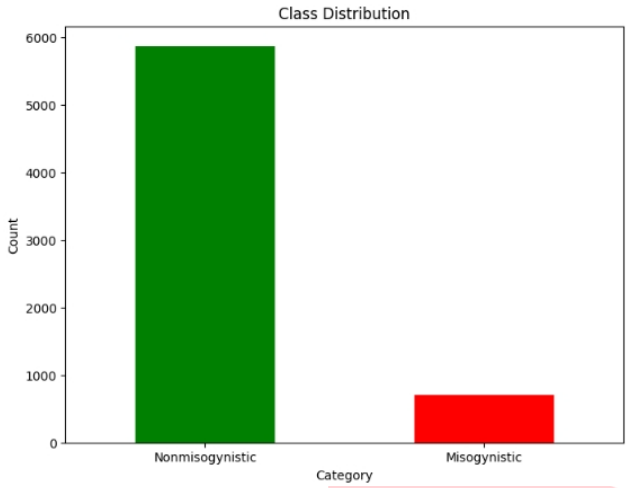
B. Persiapan Dataset

Dataset yang akan digunakan diambil dari penelitian yang dilakukan oleh Ella Guest dkk. [9] dengan judul "*An Expert Annotated Dataset for the Detection of Online Misogyny*". Dataset disediakan dalam bentuk *spreadsheet* dengan format *Comma Separated Values* (CSV). Menurut penelitian Ella, Data tersebut diambil dari Februari 2020 hingga Mei 2020. Data yang tersedia meliputi 6567 data berisi komentar di Reddit beserta informasi yang berkaitan dengan komentar tersebut. Setiap komentar telah diberi label yang menunjukkan bahwa komentar tersebut bersifat misoginistik atau non misoginistik. Berikut pada tabel 1 merupakan contoh isi dataset mentah, isinya hanya berisi 2 kolom dengan data yang akan digunakan yaitu kolom “body” yang berisi komentar di Reddit yang mungkin memiliki konten misogini lalu kolom “level_1” yang berisi label yang menunjukkan bahwa teks pada kolom “body” bersifat *Misogynistic* atau *Nonmisogynistic*.

TABEL 1
(Contoh Data Mentah)

index	body	level_1
0	Do you have the skin of a 80 year old grandma? Worry no more, just drink water!	Nonmisogynistic
1	"This is taking a grain of truth and extrapolating to insanity. Stay hydrated, it's healthy, you'll look and feel better. It will not reverse the aging process though."	Nonmisogynistic
2	Honestly my favorite thing about this is that they feel the need to cite beauty professionals in order to prove that dehydration is caused by not drinking enough water.	Nonmisogynistic

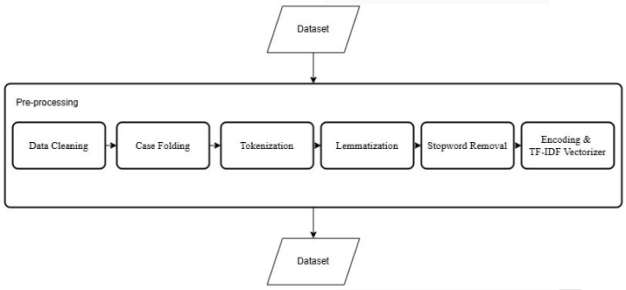
Analisis pada dataset dengan melakukan perhitungan distribusi kelas menunjukkan adanya ketidakseimbangan data. Perhitungan jumlah data berdasarkan label misogini yang diberikan pada setiap data menunjukkan bahwa dataset memiliki data non-misogini lebih banyak (89.35%) dibandingkan data misogini (10.64%). Dalam klasifikasi, ketidakseimbangan data seperti ini dapat menyebabkan adanya bias terhadap kelas mayoritas sehingga menghasilkan performa yang buruk untuk kelas minoritas. Suatu model dapat memiliki hasil yang baik tapi tidak dapat dianggap memiliki performa yang baik karena model tersebut tidak akan dapat mendeteksi kelas minoritas dengan benar [13]. Distribusi kelas dapat dilihat pada gambar 2.



GAMBAR 2
(Distribusi Kelas)

C. Tahap Preprocessing

Dataset yang didapatkan merupakan data mentah yang belum layak digunakan untuk proses klasifikasi. Oleh karena itu, dilakukan tahap preprocessing untuk membersihkan data. Tahap preprocessing merupakan tahap yang dilakukan untuk membersihkan data teks mentah dan membuatnya menjadi lebih terstruktur sehingga meningkatkan kualitas pengujian dengan hasil yang akurat. Alur tahapan penelitian dapat dilihat pada gambar 3.



GAMBAR 3
(Tahapan Preprocessing)

Berikut merupakan penjelasan setiap tahap yang dilakukan dalam preprocessing:

a. Data Cleaning

Pada tahap ini, data dibersihkan dengan menghapus kolom yang tidak relevan, mengubah nama kolom "level_1" menjadi "category", menghapus data duplikat, dan mengubah tipe data sehingga dapat dipakai pada proses lainnya.

b. Case Folding

Pada tahap ini, semua huruf pada teks diubah menjadi huruf non-kapital untuk memastikan uniformitas data.

c. Tokenization

Pada tahap ini, data teks yang merupakan kalimat dipisahkan teks menjadi kata-kata individu.

d. Lemmatization

Pada tahap ini, kata-kata diubah menjadi menjadi bentuk dasarnya dengan menghapus imbuhan yang ada.

e. Stopword Removal

Pada tahap ini, dilakukan penghapusan stopwords yang berarti kata-kata umum yang tidak terlalu penting seperti "and", "the", "is", dan sebagainya.

f. Encoding

Pada tahap ini, label pada kolom "category" diubah menjadi numerik sehingga dapat lebih mudah dibaca oleh komputer.

g. TF-IDF Vectorization

Pada tahap ini, teks pada kolom "body" diubah menjadi numerik sehingga dapat lebih mudah dibaca oleh komputer.

Berikut pada tabel 2 merupakan hasil dataset yang telah melalui proses preprocessing.

TABEL 2
(Contoh hasil data teks pada kolom body setelah melalui tahap preprocessing)

Proses	Output
Original Text	I never understood this part of male psychology, like for me seeing a hot guy in a picture doesnt make me any more or less inclined to upvote/like a picture. What exactly is it about men that makes them more inclined to upvote pics with stacy in them? Like are they just EXTREMELY horny all the time? If so Im ngl I underestimated their horniness like I wouldnt think it would affect stuff like this
Data Cleaning	I never understood this part of male psychology, like for me seeing a hot guy in a picture doesnt make me any more or less inclined to upvote/like a picture. What exactly is it about men that makes them more inclined to upvote pics with stacy in them? Like are they just EXTREMELY horny all the time? If so Im ngl I underestimated their horniness like I wouldnt think it would affect stuff like this
Case Folding	i never understood this part of male psychology, like for me seeing a hot guy in a picture doesnt make me any more or less inclined to upvote/like a picture. what exactly is it about men that makes them more inclined to upvote pics with stacy in them? like are they just extremely horny all the time? if so im ngl i underestimated their horniness like i wouldnt think it would affect stuff like this
Tokenization	'i', 'never', 'understood', 'this', 'part', 'of', 'male', 'psychology', ',', 'like', 'for', 'me', 'seeing', 'a', 'hot', 'guy', 'in', 'a', 'picture', 'doesnt', 'make', 'me', 'any', 'more', 'or', 'less', 'inclined', 'to', 'upvote/like', 'a', 'picture', '.', 'what', 'exactly', 'is', 'it', 'about', 'men', 'that', 'makes', 'them', 'more', 'inclined', 'to', 'upvote', 'pics', 'with', 'stacy', 'in', 'them', '?', 'like', 'are', 'they', 'just', 'extremely', 'horny', 'all', 'the', 'time', '?', 'if', 'so', 'im', 'ngl', 'i', 'underestimated', 'their', 'horniness', 'like', 'i', 'wouldnt', 'think', 'it', 'would', 'affect', 'stuff', 'like', 'this'
Lemmatization	'i', 'never', 'understand', 'this', 'part', 'of', 'male', 'psychology', ',', 'like', 'for', 'me', 'see', 'a', 'hot', 'guy', 'in', 'a', 'picture', 'doesnt', 'make', 'me', 'any', 'more', 'or', 'less', 'inclined', 'to', 'upvote/like', 'a', 'picture', '.', 'what', 'exactly', 'be', 'it', 'about', 'men', 'that', 'make', 'them', 'more', 'inclined', 'to', 'upvote', 'pic', 'with', 'stacy', 'in', 'them', '?', 'like', 'be', 'they', 'just', 'extremely', 'horny', 'all', 'the', 'time', '?', 'if', 'so', 'im', 'ngl', 'i', 'underestimate', 'their', 'horniness', 'like', 'i',

Proses	Output
	'wouldnt', 'think', 'it', 'would', 'affect', 'stuff', 'like', 'this'
Stop Words Removal	'never', 'understand', 'part', 'male', 'psychology', 'like', 'see', 'hot', 'guy', 'picture', 'doesnt', 'make', 'less', 'inclined', 'picture', 'exactly', 'men', 'make', 'inclined', 'upvote', 'pic', 'stacy', 'like', 'extremely', 'horny', 'time', 'im', 'ngl', 'underestimate', 'horniness', 'like', 'wouldnt', 'think', 'would', 'affect', 'stuff', 'like'

Setelah itu dilakukan proses *encoding* untuk mengubah label "Nonmisogynistic" dan "Misogynistic" pada pada kolom "level_1", yang diubah namanya menjadi kolom "category", menjadi numerik agar dapat lebih mudah dibaca oleh komputer. Hasilnya yaitu angka 1 merepresentasikan kelas "Nonmisogynistic" dan 0 merepresentasikan kelas "Misogynistic". Tabel 3 berikut merupakan hasil dari proses *encoding*.

TABEL 3
(Contoh dataframe hasil preprocessing)

index	body	category
0	skin year old grandma worry drink water	1
1	take grain truth extrapolate insanity stay hydrate healthy look feel well reverse age process though	1
2	honestly favorite thing feel need cite beauty professional order prove dehydration cause drink enough water	1
3	source doesnt sound right idk	1
4	damn saw movie old woman bath blood virgin one tell need water	0

Selanjutnya dilakukan vektorisasi TF-IDF untuk mengubah teks pada kolom "body" menjadi bentuk numerik agar dapat dibaca oleh komputer. Cara kerja vektorisasi TF-IDF yaitu melakukan perhitungan bobot pada kata-kata yang lebih informatif atau lebih penting dalam sebuah dokumen atau corpus. Gambar 4 menunjukkan hasil vektorisasi TF-IDF yang dilakukan.

```
Feature names:  ['aa' 'abandon' 'ability' ... 'zero' 'zone' 'zuparic']
TF-IDF Result:
  (0, 1370)     0.43414235442252236
  (0, 3091)     0.3415478078130002
  (0, 4041)     0.4866382529201635
  (0, 4831)     0.4392034798722779
  (0, 4941)     0.4264133995942537
  (0, 4978)     0.28865314442071577
  (1, 106)      0.26606037971059576
  (1, 1657)     0.18578641391105555
  (1, 1905)     0.4022227663150848
  (1, 2014)     0.27906388228628043
```

GAMBAR 4
(Contoh hasil TF-IDF)

Pada gambar 4, "Feature names" berisi semua kata unik yang dipelajari TF-IDF Vectorizer dari semua dokumen dalam data. Hasil yang ditampilkan diurutkan berdasarkan nomor indeks kata dan sesuai urutan alfabet A ke Z, sehingga kata 'aa' memiliki nomor indeks 0. Dibawahnya, terdapat hasil TF-IDF yang diurutkan secara naik berdasarkan indeks dokumen. Sehingga, sebagai contoh yaitu hasil pertama berarti dalam indeks dokumen 0, kata dengan nomor indeks 1370 memiliki nilai TF-IDF 0.43 yang merupakan nilai pentingnya kata tersebut pada dokumen.

D. Support Vector Machine (SVM)

SVM merupakan algoritma *machine learning* yang dapat digunakan untuk klasifikasi dan regresi. Algoritma ini sangat efektif untuk data yang dengan dimensi tinggi dan memiliki hubungan nonlinier. Tujuan utama penggunaannya adalah untuk menemukan *hyperplane* optimal yang dapat memisahkan titik-titik data menjadi kelas yang berbeda. Algoritma ini memaksimalkan margin antara titik terdekat setiap kelas yang berbeda, atau disebut juga dengan *support vector*. Dalam klasifikasi teks, SVM mengubah data teks menjadi fitur numerik dan juga mengidentifikasi *hyperplane* terbaik untuk memisahkan berbagai kategori teks. Untuk data yang tidak dapat dipisahkan secara linier, SVM menggunakan fungsi kernel untuk mengubah data menjadi ruang berdimensi lebih tinggi sehingga dapat dipisahkan secara linier [14].

Untuk melakukan klasifikasi teks pada penelitian ini, model berbasis SVM yang digunakan yaitu *Support Vector Classification* (SVC). SVC dipilih sebagai model yang digunakan untuk merepresentasikan SVM karena versatilitasnya, efektivitasnya dalam ruang dimensi tinggi, efisiensi memori, dan fleksibilitas nya. Sebagai model yang serbaguna, SVC dapat menangani masalah klasifikasi linier dan non-linier dengan menggunakan berbagai fungsi kernel (misalnya, *Linear*, *Polinomial*, *Radial Basis Function* (RBF), *Sigmoid*, dan *Precomputed*). SVC efektif dalam ruang dimensi tinggi, sehingga cocok untuk klasifikasi teks dimana jumlah fitur (kata) bisa sangat besar. SVC menggunakan subset titik pelatihan (vektor pendukung) dalam fungsi keputusan, yang membuatnya efisien dalam penggunaan memori. Fungsi kernel yang berbeda dapat ditentukan untuk fungsi keputusan, memungkinkan penyesuaian berdasarkan masalah yang dihadapi. Disamping berbagai kelebihan yang dimiliki SVC, model ini juga memiliki kekurangan yaitu sensitivitasnya terhadap data yang tidak seimbang sehingga dapat menunjukkan bias terhadap kelas mayoritas jika tidak disetel dengan baik [15][16].

Pada penelitian ini, model SVC yang digunakan berasal dari library *scikit-learn* yang secara *default* menggunakan kernel RBF. Kernel tersebut cocok digunakan karena fleksibel dan dapat mengatasi relasi non-linear antar fiturnya seperti relasi antar kata pada teks yang digunakan untuk klasifikasi dan juga kemampuannya untuk beroperasi menggunakan data berdimensi tinggi seperti data yang telah diproses menggunakan TF-IDF[17].

E. Stochastic Gradient Descent (SGD)

SGD adalah metode iteratif untuk mengoptimalkan parameter model dengan menggunakan subset acak dari data training, sehingga membuatnya efisien untuk kumpulan data yang besar dan pembelajaran secara real-time. Proses SGD melibatkan inisialisasi parameter, pengacakan data, penghitungan gradien, dan pembaruan parameter untuk meminimalkan *cost function*. Dalam klasifikasi teks, metode klasifikasi SGD membantu menemukan *decision boundary* yang optimal untuk memisahkan titik-titik data, dan biasanya menggunakan *cross-entropy loss*. Keuntungan utama dari SGD yaitu dapat diukur, efisien, dan dapat menghindari

minimum lokal karena unsur keacakan yang ada di dalamnya [18].

Untuk melakukan klasifikasi teks pada penelitian ini, model berbasis SGD yaitu *Stochastic Gradient Descent Classifier* (SGDClassifier). SGDClassifier merupakan algoritma klasifikasi linear berbasis SGD, biasanya digunakan untuk klasifikasi biner dan klasifikasi *multi-class*. Karakteristiknya berupa pendekatannya secara iteratif untuk menyesuaikan parameter model untuk meminimalkan cost function dengan menggunakan cross-entropy loss. Model klasifikasi ini cocok digunakan untuk masalah machine learning yang berskala besar dan terbatas, sehingga sangat baik digunakan dalam klasifikasi teks dan identifikasi gambar. SGDClassifier bekerja dengan memperbarui parameter model menggunakan subset yang dipilih secara acak dari data pelatihan di setiap iterasi, bukan seluruh dataset. Pendekatan SGD ini memungkinkan model untuk konvergen lebih cepat dibandingkan dengan metode gradient descent tradisional. Dengan memproses kumpulan data yang lebih kecil, model ini dapat melakukan pembaruan bertahap pada parameter model, yang membantu menangani dataset yang besar secara efisien dan memungkinkan pembelajaran secara real-time.

Keunggulan yang dimiliki SGDClassifier adalah efisiensinya untuk dataset besar karena melakukan training dengan sampel kecil pada satu waktu, dapat beradaptasi dan belajar dengan data yang masuk secara real time, konvergensi cepat dikarenakan pembaruan parameter yang lebih sering, dan dukungan regularisasi L1 dan L2 yang membantu mencegah *overfitting* untuk data berdimensi tinggi. Walaupun begitu, SGDClassifier juga memiliki kekurangan berupa sifat stokastiknya yang dapat menyebabkan konvergensi lebih lambat atau solusi yang kurang optimal, penyetelan laju pembelajarannya yang sangat penting tapi juga sulit dilakukan, sensitivitasnya terhadap penskalaan fitur, dan kemampuan pemodelan yang terbatas disebabkan oleh karakteristiknya sebagai pengklasifikasi linear [19].

Pada penelitian ini, model SGDClassifier yang digunakan berasal dari library *scikit-learn* tanpa menggunakan parameter *loss* sehingga secara *default* akan menggunakan fungsi *hinge loss.* Penggunaan fungsi tersebut berarti model ini melatih model linearSVM menggunakan *L2 regularization* yang dapat mencegah overfitting dan juga menggunakan optimasi berbasis SGD yang membuatnya baik digunakan terhadap data yang terpisah secara linear dan efektif untuk data berdimensi tinggi [20].

F. Mengatasi Class Imbalance

Berdasarkan perhitungan distribusi kelas sebelumnya, didapatkan bahwa dataset yang akan digunakan memiliki *Class Imbalance* atau ketidakseimbangan kelas. Pada penelitian ini, teknik yang akan digunakan untuk mengatasi ketidakseimbangan kelas yaitu dengan melakukan *Oversampling* dan *Undersampling*, atau disebut juga dengan *Resampling*.

Pada teknik *Oversampling*, jumlah kelas minoritas ditambah dengan menduplikasi data yang ada atau membuat data baru. Disini, metode yang digunakan untuk *Oversampling* adalah *Synthetic Minority Oversampling Technique* (SMOTE). Dengan menggunakan teknik SMOTE, sampel data baru dibuat dengan melakukan interpolasi antara sampel data minoritas yang ada. Teknik ini dapat meningkatkan *Recall* pada kelas minoritas karena memberikan lebih banyak sampel yang dapat dipelajari oleh model, tapi teknik ini juga dapat menyebabkan tingkat *false positive* yang lebih tinggi dikarenakan sampel sintetik yang dibuat tidak merepresentasikan distribusi asli kelas minoritas.

Pada teknik *Undersampling*, jumlah kelas mayoritas dikurangi sampai menyeimbangkan dataset. Metode yang digunakan untuk *Undersampling* adalah RandomUnderSampler. Metode *Random UnderSampling* ini akan mengurangi jumlah sampel data dari kelas mayoritas secara acak. Teknik ini dapat meningkatkan kemampuan model untuk mendeteksi kelas minoritas dengan mengurangi bias terhadap kelas mayoritas, tapi teknik ini juga dapat menyebabkan kehilangan informasi disebabkan oleh penghapusan data sehingga dapat mengurangi akurasi klasifikasi secara keseluruhan [21].

G. K-Fold Cross Validation

K-Fold Cross Validation merupakan metode pembagian data yang digunakan untuk mengurangi bias pada data dengan menggunakan sebagian data sebagai subset yang lalu dirotasi berdasarkan jumlah K yang ditetapkan [22]. Pada pengujian pertama, subset S1 digunakan sebagai data test sedangkan sisanya digunakan sebagai data training. Pada pengujian selanjutnya, subset S2 digunakan sebagai data test dan subset S1 bergabung dengan sisa data menjadi data training. Pada penelitian ini, jumlah K yang digunakan yaitu K = 5.

H. Metrik Evaluasi

Evaluasi dilakukan dengan menggunakan *Confusion Matrix* untuk mendapatkan nilai performa model berdasarkan jumlah K yang digunakan pada *K-Fold Cross Validation*. Nilai yang didapatkan yaitu nilai *True Positive* (TP), *False Positive* (FP), *False Negative* (FN), dan *True Negative* (TN). Nilai tersebut lalu digunakan untuk menghitung performa model dalam bentuk *Precision*, *Recall*, *F1-Score*, dan *Accuracy* [23]. Perhitungan dilakukan untuk kelas positif yaitu kelas yang digunakan sebagai tujuan klasifikasi teks yang benar memiliki konten misogini yang berarti kelas 1 dengan label “Misogynistic”. Berikut adalah rumus-rumus yang digunakan:

a. Precision

Precision mengukur rasio seberapa banyak teks yang diidentifikasi positif adalah benar positif. Nilai *Precision* tinggi berarti model dapat menghindari *false positive* sehingga dapat memprediksi dengan kemungkinan tinggi benar. Nilai *Precision* rendah sebaliknya menunjukkan model menghasilkan prediksi *false positive* yang tinggi sehingga dapat memberi label positif untuk teks yang sebenarnya negatif.

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (1)$$

b. Recall

Recall, disebut juga sensitivitas atau ukuran *true positive*, mengukur rasio seberapa banyak teks positif yang dapat diidentifikasi dengan benar oleh model. Nilai *Recall* yang tinggi menunjukkan bahwa model sangat baik dalam menemukan teks yang benar positif. Nilai *Recall* yang rendah

sebaliknya menunjukkan bahwa model kesulitan menemukan teks yang benar positif.

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

c. F1-Score

F1-Score mengukur rata-rata harmonis antara Precision dan Recall, digunakan untuk mengukur keseimbangan antara Precision dan Recall.

$$F1\text{–}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \times 100\% \quad (3)$$

d. Accuracy

Accuracy mengukur rasio instansi yang diprediksi dengan benar (untuk true positive dan true negative) terhadap total instansi keseluruhan. Nilai *Accuracy* yang tinggi menunjukkan bahwa model secara keseluruhan dapat melakukan klasifikasi dengan benar. Nilai *Accuracy* yang rendah menunjukkan bahwa model memiliki banyak kesalahan pada prediksinya.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (4)$$

IV. HASIL DAN PEMBAHASAN

Klasifikasi dilakukan dengan menggunakan SVC untuk merepresentasikan metode SVM dan SGDClassifier untuk metode SGD. Kedua metode klasifikasi mengikuti tahap yang sama, yaitu dengan melatih model menggunakan data training lalu mengevaluasi performanya pada data test. Pembagian data dilakukan dengan menggunakan metode *K-Fold Cross Validation* dengan K = 5 sehingga pembagian data yang dihasilkan yaitu sebesar 80% untuk data training dan 20% untuk data testing. Untuk mengatasi ketidakseimbangan data, maka dilakukan dua klasifikasi tambahan untuk setiap model dimana salah satunya menggunakan metode *Oversampling* dan yang lainnya menggunakan *Undersampling*. Ini dilakukan untuk mendapatkan hasil terbaik antara kedua metode *Resampling* pada kedua model klasifikasi. Hasil yang didapatkan berupa nilai *Accuracy*, *Precision*, *Recall*, dan *F1-Score* yang dapat digunakan sebagai metrik evaluasi hasil klasifikasi. Tabel 4 dan 5 berisi hasil klasifikasi menggunakan SVC dan tabel 6 dan 7 berisi hasil klasifikasi menggunakan SGDClassifier.

TABEL 4
(Hasil Klasifikasi SVC)

	Fold (k)	Accuracy	Precision	Recall	F1-Score
SVC	1	92.20%	**100%**	7%	13%
	2	92.28%	90%	9%	16%
	3	91.80%	75%	3%	6%
	4	92.28%	89%	8%	14%
	5	**92.68%**	**100%**	**12%**	**21%**
SVC + Oversampling	1	90.50%	27%	8%	12%
	2	**90.58%**	**35%**	**15%**	**21%**
	3	89.44%	15%	6%	8%
	4	90.00%	29%	14%	19%
	5	90.57%	34%	**15%**	**21%**
SVC + Undersampling	1	**84.57%**	**32%**	74%	**44%**
	2	81.80%	28%	**75%**	41%
	3	81.72%	26%	62%	36%
	4	82.28%	28%	71%	40%
	5	83.82%	31%	**75%**	**44%**

TABEL 5
(Rata-rata hasil SVC)

		Accuracy	Precision	Recall	F1-Score
SVC		**92.25%**	**91%**	8%	14%
SVC + Oversampling		90.22%	28%	12%	16%
SVC + Undersampling		82.84%	29%	**71%**	**41%**

Berdasarkan hasil yang ditampilkan pada tabel 4 dan rata-ratanya pada tabel 5, didapatkan bahwa model SVC dapat menghasilkan *Accuracy* dan *Precision* yang tinggi, namun memiliki *Recall* yang rendah sehingga menghasilkan nilai *F1-Score* yang rendah. Pada penggunaan *Oversampling*, terdapat sedikit penurunan *Accuracy* dan kenaikan *Recall*, namun *Precision* menurun dengan signifikan, hasilnya nilai *F1-Score* yang didapatkan hanya sedikit lebih baik dari SVC tanpa *Resampling*. Pada penggunaan *Undersampling*, terdapat penurunan *Accuracy* dan *Precision* yang mirip dengan penggunaan *Oversampling*, namun terdapat kenaikan *Recall* yang signifikan, menghasilkan nilai *F1-Score* tertinggi untuk model SVC.

TABEL 6
(Hasil Klasifikasi SGDClassifier)

	Fold (k)	Accuracy	Precision	Recall	F1-Score
SGDClassifier	1	93.34%	**84%**	25%	39%
	2	93.10%	74%	27%	40%
	3	92.36%	62%	23%	34%
	4	93.09%	72%	27%	40%
	5	**93.50%**	81%	**28%**	**42%**
SGDClassifier + Oversampling	1	76.44%	19%	**55%**	28%
	2	**85.95%**	**29%**	49%	**37%**
	3	74.82%	18%	**55%**	27%
	4	81.46%	23%	51%	31%
	5	79.84%	21%	52%	30%
SGDClassifier + Undersampling	1	**76.52%**	**22%**	73%	**34%**
	2	73.76%	20%	73%	32%
	3	74.57%	20%	66%	30%
	4	72.60%	19%	**74%**	31%
	5	74.31%	21%	**74%**	32%

TABEL 7
(Rata-rata hasil SGDClassifier)

		Accuracy	Precision	Recall	F1-Score
SVC		**93.08%**	**75%**	26%	**39%**
SVC + Oversampling		79.70%	22%	52%	31%
SVC + Undersampling		74.35%	20%	**72%**	32%

Berdasarkan hasil yang ditampilkan pada tabel 6 dan rata-ratanya pada tabel 7, didapatkan hasil untuk model SGDClassifier dengan *Accuracy* dan *Precision* yang cukup tinggi namun memiliki *Recall* yang rendah, menghasilkan *F1-Score* yang rendah. Dengan penggunaan *Oversampling* dan *Undersampling*, SGDClassifier mendapatkan *Accuracy* dan *Precision* yang lebih rendah namun *Recall* naik secara signifikan, mirip seperti penggunaannya pada SVC sebelumnya. Namun, berbeda dengan SVC, penggunaan *Oversampling* dan *Undersampling* pada SGDClassifier tidak menghasilkan *F1-Score* yang lebih tinggi, sehingga hasilnya SGDClassifier tanpa *Resampling* memiliki nilai *F1-Score* tertinggi untuk pengujian SGDClassifier.

Secara keseluruhan, rata-rata hasil terbaik didapatkan oleh model SVC dengan menggunakan *Undersampling* dengan *Accuracy* 82.84%, *Precision* 29%, *Recall* 71%, dan *F1-Score* 41%. Meskipun demikian, hasilnya tidak berbeda jauh dari rata-rata hasil terbaik untuk model SGDClassifier tanpa menggunakan teknik *Resampling* dengan hasilnya *Accuracy* 93.08%, *Precision* 75%, *Recall* 26%, dan *F1-Score* 39%. Hasil ini menunjukkan karakteristik SVC yang menunjukkan efektivitasnya dalam menangani data dengan ruang dimensi tinggi dengan memiliki *Accuracy* yang tinggi, namun memiliki kekurangan berupa sensitivitasnya terhadap data yang tidak seimbang menghasilkan nilai *Recall* yang rendah untuk kelas minoritas sehingga membutuhkan teknik *Resampling* untuk meningkatkan performanya. Untuk model SGDClassifier, model ini bekerja secara iteratif dan memiliki skalabilitas terhadap dataset yang besar. Karakteristik ini membuat SGD dapat menyesuaikan terhadap distribusi data yang tidak seimbang sehingga menghasilkan performa keseluruhan yang lebih baik tanpa menggunakan teknik *Resampling*.

Sebagai hasil klasifikasi *baseline*, Ella Guest dkk. [6] telah melakukan klasifikasi pada dataset yang sama dan mendapatkan hasil terbaik menggunakan model BERT yang berbobot dengan hasil *Accuracy* 93%, *Precision* 67%, *Recall* 30%, dan *F1-Score* 42%. Jika dibandingkan dengan hasil terbaik pada penelitian ini, yaitu menggunakan model SVM dengan teknik *Undersampling*, maka didapatkan bahwa hasilnya tidak lebih baik dari *baseline*. Hasil klasifikasi pada penelitian ini juga memiliki masalah yang sama seperti *baseline* dimana hasilnya berupa *Accuracy* dan *Precision* yang tinggi namun memiliki *Recall* yang rendah. Hasil ini disebabkan oleh dataset yang tidak seimbang. Setelah melalui proses *Resampling*, *Recall* mengalami kenaikan, tetapi sebagai gantinya *Accuracy* dan *Precision* menurun.

V. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa untuk klasifikasi pada dataset berisi 6.567 konten misogini dan non-misogini pada reddit dengan menggunakan model klasifikasi SVC yang berbasis SVM dan SGDClassifier yang berbasis SGD, kedua model memiliki kesulitan dalam mendeteksi konten misogini. Secara keseluruhan, model SVC dengan teknik *Undersampling* menghasilkan performa yang lebih baik dibandingkan dengan SGDClassifier dalam mengidentifikasi konten misogini dengan model SVC dengan teknik *Resampling* memberikan hasil terbaik yaitu *Accuracy* 82.84%, *Precision* 29%, *Recall* 71%, dan *F1-Score* 41%. Hasil penelitian ini menunjukkan bahwa model SVC dapat mengklasifikasi data teks dengan konten misogini lebih baik dibandingkan dengan model SGDClassifier namun SVC memerlukan teknik *Resampling* untuk mengatasi ketidakseimbangan data untuk mendapatkan hasil terbaik sedangkan SGDClassifier dapat memiliki hasil terbaiknya tanpa memerlukan teknik *Resampling*. Meskipun begitu, kedua hasil terbaik dari penelitian ini tidak lebih baik jika dibandingkan dengan hasil klasifikasi *baseline*.

REFERENSI

[1] S. Kemp, “Digital 2022: Indonesia,” DataReportal, 15 Feb. 2022. [Daring]. Tersedia: https://datareportal.com/reports/digital-2022-indonesia. [Diakses: 18-Jan-2025].

[2] “Social Media,” Cambridge Dictionary, [Daring]. Tersedia: https://dictionary.cambridge.org/dictionary/english/social-media. [Diakses: 18-Jan-2025].

[3] “Online Abuse,” Get Safe Online, [Daring]. Tersedia: https://www.getsafeonline.org/personal/articles/online-abuse/. [Diakses: 18-Jan-2025].

[4] R. S. Angeline, D. Nurjanah, dan H. Nurrahmi, “Misogyny speech detection using Long Short-Term Memory and BERT embeddings,” *2022 5th International Conference on Information and Communications Technology (ICOIACT)*, hlm. 155–159, Aug. 2022, doi: 10.1109/icoiact55506.2022.9972171.

[5] Komnas Perempuan, “Catatan Tahunan Komnas Perempuan Tahun 2020: Perempuan dalam Himpitan Pandemi: Lonjakan Kekerasan Seksual, Kekerasan Siber, Perkawinan Anak, dan Keterbatasan Penanganan di Tengah Covid-19,” Jakarta, 5 Maret 2021. [Daring]. Tersedia: https://komnasperempuan.go.id/uploadedFiles/1463.1614929011.pdf

[6] B. Tri Wibowo, D. Nurjanah, dan H. Nurrahmi, "Identification of Misogyny on Social Media in Indonesian Using Bidirectional Encoder Representations From Transformers (BERT)," *2023 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, Bali, Indonesia, 2023, hlm. 401-406, doi: 10.1109/ICAIIC57133.2023.10067106.

[7] L. G. Moreno-Sandoval, A. Pomares-Quimbaya, S. A. Barbosa-Sierra, dan L. M. Pantoja-Rojas, “Detection of hate speech, racism and misogyny in digital social networks: Colombian case study,” *Big Data and Cognitive Computing*, vol. 8, no. 9, hlm. 113, Sep. 2024, doi: 10.3390/bdcc8090113.

[8] H. Liu, F. Chiroma, dan M. Cocea, “Identification and classification of misogynous tweets using multi-classifier fusion,” *IberEval@SEPLN*, hlm. 268–273, Jul. 2018, [Daring]. Tersedia: http://ceur-ws.org/Vol-2150/AMI_paper7.pdf.

[9] E. Guest, B. Vidgen, A. Mittos, N. Sastry, G. Tyson, dan H. Margetts, “An Expert Annotated Dataset for the Detection of Online Misogyny,” *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, hlm. 1336–1350, Apr. 2021, doi: 10.18653/v1/2021.eacl-main.114.

[10] R. Umar, I. Riadi, dan P. Purwono, “Comparison of SVM, RF and SGD Methods for Determination of Programmer's Performance Classification Model in Social Media Activities,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 2, hlm. 329–335, 2020, doi: 10.29207/resti.v4i2.1770.

[11] I. Oktanisa dan A. A. Supianto, “Perbandingan teknik klasifikasi dalam data mining untuk bank Direct Marketing,” *Jurnal Teknologi Informasi Dan Ilmu Komputer*, vol. 5, no. 5, hlm. 567–576, Oktober 2018, doi: 10.25126/jtiik.201855958.

[12] A. B. Prasetijo, R. R. Isnanto, D. Eridani, Y. A. A. Soetrisno, M. Arfan, dan A. Sofwan, “Hoax detection system on Indonesian news sites based on text classification using SVM and SGD,” *2017 4th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, hlm. 45–49, Oktober 2017, doi: 10.1109/icitacee.2017.8257673.

[13] J. Brownlee, “A Gentle Introduction to Imbalanced Classification,” Machine Learning Mastery, 14 Jan. 2020. [Daring]. Tersedia: https://machinelearningmastery.com/what-is-imbalanced-classification/. [Diakses: 18-Jan-2025].

[14] “Support Vector Machine (SVM) Algorithm,” GeeksforGeeks, 10 Okt. 2024. [Daring]. Tersedia: https://www.geeksforgeeks.org/support-vector-machine-algorithm/. [Diakses: 18-Jan-2025].

[15] J. Joseph, “SVC in Machine Learning: What You Need to Know,” reason.town, 15 Agustus 2022. [Daring]. Tersedia: https://reason.town/svc-in-machine-learning/. [Diakses: 18-Jan-2025].

[16] N. L. P. M. S. Putri Waisnawa, D. Nurjanah, dan H. Nurrahmi, “Cyberbullying Detection on Twitter using Support Vector Machine Classification Method,” *Building of Informatics, Technology and Science (BITS)*, vol. 3, no. 4, hlm. 661, Maret 2022, doi: 10.47065/bits.v3i4.1435.

[17] “Support Vector Classification (SVC),” scikit-learn documentation, [Daring]. Tersedia: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html#sklearn.svm.SVC. [Diakses: 18-Jan-2025].

[18] F. T. Admojo dan Y. I. Sulistya, “Analisis performa algoritma Stochastic Gradient Descent (SGD) dalam mengklasifikasi tahu berformalin,” *Indonesian Journal of Data and Science*, vol. 3, no. 1, hlm. 1–8, Mar. 2022, doi: 10.56705/ijodas.v3i1.42.

[19] “Stochastic Gradient Descent Classifier,” GeeksforGeeks, 04 Nov. 2023. [Daring]. Tersedia: https://www.geeksforgeeks.org/stochastic-gradient-descent-classifier/. [Diakses: 18-Jan-2025].

[20] “SGDClassifier,” scikit-learn documentation, [Daring]. Tersedia: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html. [Diakses: 18-Jan-2025].

[21] “How to Handle Imbalanced Classes in Machine Learning,” GeeksforGeeks, 18 Mar. 2024. [Daring]. Tersedia: https://www.geeksforgeeks.org/how-to-handle-imbalanced-classes-in-machine-learning/. [Diakses: 18-Jan-2025].

[22] V. Tamuntuan, K. Kusrini, dan K. Kusnawi, “Analisis Perbandingan Kinerja Algoritma Klasifikasi Pada Mahasiswa Berpotensi Dropout,” *Building of Informatics Technology and Science (BITS)*, vol. 6, no. 2, hlm. 847-855, Sep. 2024, doi: 10.47065/bits.v6i2.5658.

[23] A. Putri, A. Aryanti, dan S. Soim, “Implementasi Algoritma SVM Non-Linear Pada Klasifikasi Analisis Sentimen Perkembangan AI di Sektor Pendidikan,” *Building of Informatics Technology and Science (BITS)*, vol. 6, no. 2, hlm. 703-711, Sep. 2024, doi: 10.47065/bits.v6i2.5522.