Klasifikasi Komentar Toxic Pada Sosial Media Menggunakan SVM, Information Gain dan TF-IDF

1st Muhammad Ilham Maulana
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
ilhaammaulana@student.telkomuniversity.ac.id

2nd Kemas Muslim
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
kemasmuslim@telkomuniversity.ac.id

3rd Mahendra Dwifebri
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
mahendrap@telkomuniversity.ac.id

***Abstrak* — Sosial media merupakan suatu bentuk perantara interaksi sosial secara online. Aplikasi media sosial pun sudah dalam banyak bentuk dan di dalam sosial media ini meskipun banyak hal positif yang dapat diambil, ada beberapa juga hal-hal negatif contoh nya toxic comment. Toxic comment sendiri tidaklah mudah untuk dideteksi secara manual, maka penelitian berencana untuk mengklasifikasikan toxic comment tersebut menggunakan machine learning. Beberapa penelitian untuk klasifikasi toxic comment sudah dilakukan, dalam beberapa penelitian tersebut digunakan metode Support Vector Machine. Dalam penelitian ini metode yang digunakan adalah Support Vector Machine (SVM) sebagai classifier, Information Gain sebagai feature selection dan TF- IDF sebagai feature extraction. Data-data yang dikumpulkan adalah melalui cuitan twitter beberapa pengguna di media sosial tersebut. Komentar-komentar tersebut dikumpulkan menjadi satu lalu diklasifikasikan menggunakan metode-metode yang sudah disebutkan.**

***Kata kunci*— Sosial media, Klasifikasi teks, Toxic comment, SVM**

I. PENDAHULUAN

A. Latar Belakang

Dalam perkembangan teknologi yang pesat di era modern, sosial media sangat populer di kalangan masyarakat. Berdasarkan survey yang diambil dari tahun 2005 – 2015 di AS, penggunaan sosial media telah meningkat secara drastis [1]]. Di Indonesia sendiri pada tahun 2020 ini, pengguna internet sudah mencapai 175,4 juta pada bulan Januari, sedangkan pengguna media sosial sudah mencapai 160,0 juta dengan peningkatan sebanyak 12 juta (8,1%) dari tahun 2019 sampai Januari 2020 [2].

Salah satu hal yang menjadi semakin marak karena populernya media sosial adalah cyberbullying. Menurut Patchin dkk. [3] cyberbullying adalah kekerasan yang secara disengaja dan berulang kali dilakukan melalui medium berupa teks elektronik. Beberapa studi yang dilakukan juga menemukan bahwa cyberbullying sering ditemukan melalui pesan teks dan sosial media seperti Facebook atau Instagram, di sosial media sendiri itupun lebih banyak ditemukan pada bagian komentar yang dimungkinkan karena anonimitas yang dirasakan orang-orang di kolom komentar [4]. Maka cyberbullying yang dilakukan pada kolom komentar tersebut disebut dengan toxic comment. Toxic Comment adalah komentar yang kasar, tidak menghargai ataupun komentar yang membuat seseorang merasa tidak nyaman sehingga mereka meninggalkan diskusi [5].

Karena banyaknya cyberbullying melalui komentar toxic ini banyak dilakukan, maka dilakukan klasifikasi teks menggunakan machine learning untuk mengidentifikasi komentar-komentar yang terdapat di sosial media. Beberapa penelitian telah dilakukan untuk mengklasifikasikan komentar toxic menggunakan beberapa metode yang berbeda [6].

Dalam penelitian ini, metode classifier yang dipilih adalah Support Vector Machine (SVM). Metode SVM untuk mengklasifikasikan teks ini telah dibandingkan dengan metode lain, salah satunya adalah Naive Bayes. Penelitian tersebut menunjukkan bahwa metode SVM memiliki hasil f-measure yang lebih baik daripada metode Naive Bayes [7].

1. Topik dan Batasannya

Pada penelitian ini penulis melakukan klasifikasi komentar toxic dengan menggunakan metode Support Vector Machine dan feature selection Information Gain. Dataset yang digunakan berjumlah 711 diambil dari sosial media Twitter dengan label Toxic dan Non-Toxic.

2. Tujuan

Tujuan dari tugas akhir ini adalah mengidentifikasi toxic comment di sosial media dan mengetahui akurasi klasifikasi teks toxic comment menggunakan metode SVM dan Information Gain.

3. Organisasi Tulisan

Struktur penulisan dari tugas akhir ini disusun sebagai berikut: Bagian pertama berisi pendahuluan terkait tugas akhir ini. Bagian kedua menjelaskan studi yang terkait dengan tugas akhir ini. Bagian ketiga akan menjelaskan pemodelan dan performansi dari sistem yang dibangun. Bagian keempat menjelaskan hasil dan evaluasi hasil pengujian yang telah dilakukan pada bagian ketiga.

Kemudian, pada bagian terakhir menjelaskan kesimpulan dan saran berdasarkan hasil pengujian yang dilakukan pada tugas akhir ini.

II. KAJIAN TEORI

A. Studi Terkait

1. Klasifikasi Single Label

Klasifikasi Single Label adalah teknik klasifikasi yang digunakan dalam menentukan label kelas untuk data dengan suatu label L di mana jumlah L adalah lebih dari 1. Kalau L sama dengan 2 maka disebut sebagai single label classifcation atau bisa disebut juga dengan binary classification problem [8].

2. Support Vector Machine (SVM)

Support Vector Machine adalah metode pengklasifikasian supervised dengan linear kernel. Sebagai classifier dengan linear kernel, SVM bertujuan untuk mencari hyperplane yang memisahkan nilai data dengan margin maksimal di antara dua batas nilai tersebut. SVM dapat mereduksi kemungkinan overfitting karena SVM juga bertujuan untuk meminimalisasi error secara general [9].

Pada Support Vector Machine kernel digunakan untuk memetakan data masukan ke ruang dimensi yang lebih tinggi di mana batas decision dapat dibangun. Fungsi decision dapat ditulis sebagai berikut.

$$D(\mathrm{x}) = 5\emptyset(\mathrm{x}) + b \quad (1)$$

Dimana w dan b adalah parameter SVM sedangkan $w\emptyset$ adalah fungsi kernel yang memetakan data masukan ke dimensi M yang baru.

3. Infomation Gain

Information Gain adalah salah satu *feature selection* untuk klasifikasi yang cukup populer yang dapat mengukur bagus atau tidaknya suatu atribut. *Information Gain* mengukur reduksi dari entropi dengan memisahkan dataset menurut nilai yang diberikan kepada variabel acak [10]. Rumus *Information Gain* adalah sebagai berikut.

$$IG(t) = -\sum P(ci)logP(ci) + P(t)\sum P(ci|t)logP(ci|t) + P(t) \times \sum P(ci|t\ \overline{)}\ logP(ci|t$$
$$\$=1 \qquad \$=1 \qquad \$=1 \quad (2)$$

di mana ci adalah kategori ke-i, $P(ci)$ adalah probabilitas dari kategori ke-i, $P(t)$ dan $P(\overline{t})$ adalah probabilitas muncul atau tidaknya istilah t di dalam dokumen, $P(ci|t)$ adalah probabilitas muncul istilah t, sedangkan $P(ci|\overline{t})$ adalah probabilitas tidak munculnya istilah t [11].

4. TF-IDF

Jika dijelaskan secara terpisah maka Term Frequency (TF) adalah jumlah kemunculan kata dalam suatu dokumen. Inverse Document Frequency (IDF) adalah nilai dari suatu kata yang muncul. Dengan memeriksa jumlah kemunculan kata dan nilai suatu kata, maka dapat dicari kata-kata yang unik dalam dokumen tersebut. Kata-kata unik atau yang jarang ditemukan di dokumen tersebut ini memiliki nilai kata yang tinggi, sedangkan kata-kata yang sering muncul memiliki nilai rendah [12].

5. SMOTE

Masalah-masalah yang dihapadi pada klasifikasi data adalah ketidakseimbangan data. yang tidak seimbang. Contohnya pada data teks klasifikasi adalah jika salah satu label memiliki jumlah yang jauh lebih banyak maupun sedikit. Salah satu cara untuk mengatasi masalah tersebut adalah dengan melakukan oversampling atau undersampling. Ini dapat dicapai hanya dengan menduplikasi contoh dari kelas minoritas dalam dataset pelatihan sebelum menyesuaikan model. Dengan cara tersebut distribusi kelas dapat diseimbangkan tetapi tidak memberikan informasi tambahan apa pun ke model.

Teknik yang paling banyak digunakan untuk melakukan oversampling adalah Synthetic Minority Oversampling Technique atau SMOTE. Contoh cara kerja SMOTE adalah apabila ada dua label pada suatu data kelas dan salah satunya memiliki jumlah yang lebih banyak, maka data dengan label yang memiliki jumlah sedikit akan disamakan jumlahnya dengan data dengan label satunya.

6. Naive-Bayes

Naive-Bayes adalah metode klasfikasi yang menggunakan teroma Bayes. Metode Naive-Bayes mengasumsikan bahwa setiap atribut pada kelas data adalah independen. Naive-Bayes menyediakan cara untuk menghitung probabilitas setiap kelas dengan menggunakan informasi pada data sampel [14].

7. Confusion Matrix

Confusion Matrix adalah metode yang dilakukan untuk melakukan pengukuran performa suatu klasifikasi data. Ada empat istilah untuk menunjukkan hasil pada confusion matrix yaitu True Positive, True Negative, False Positive, dan False Negative. True Positive adalah keadaan di mana prediksi adalah positif dan dianggap benar, sedangkan True Negative adalah ketika hasil prediksi adalah negatif dan dianggap benar. Kemudian False Positive adalah ketika hasil prediksi adalah positif dan dianggap salah, sedangkan False Negative adalah ketika hasil prediksi adalah negatif dan dianggap salah. Untuk Confusion Matrix sendiri dapat digamabarkan ke dalam tabel berikut.

TABEL 1.
Visualisasi Confusion Matrix

	Aktual Positif	Aktual Negatif
Prediksi Positif	True Positive	True Negative
Prediksi Negatif	False Positive	False Negative

Lalu dari matriks tersebut dapat dihitung atribut-atribut berupa akurasi, presisi, *recall* dan F-1 *score.* Akurasi menggambarkan seberapa akurat model dalam mengklasifikasikan dengan benar. Akurasi dapat dihitung dengan rumus berikut.

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

Presisi menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh model. Lalu presisi dapat dihitung dengan rumus sebagai berikut.

$$Presisi = \frac{TP}{TP + FP} \quad (4)$$

Recall menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi. Untuk *recall* dapat dihitung dengan rumus sebagai berikut.

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

F-1 *score* digunakan untuk menghitung rata-rata presisi dan *recall*. Akurasi dapat kita gunakan untuk menghitung performansi ketika jumlah data yang diprediksikan seimbang, namun jika jumlahnya tidak seimbang, maka digunakanlah F1 Score sebagai acuan. Rumus F-1 *score* dapat ditulis sebagai berikut[15].

$$F1\ Score = \frac{2 \times Recall \times Presisi}{Recall + Presisi} \quad (6)$$

8. Penelitian Terkait

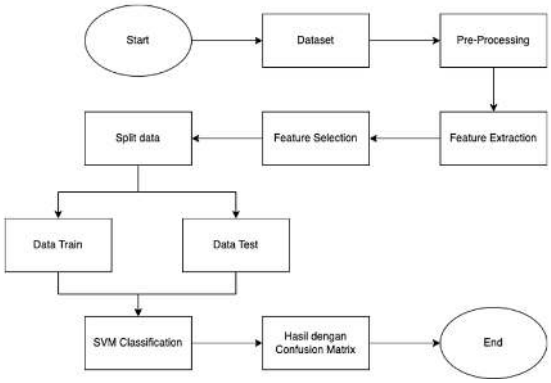
Beberapa penelitian untuk mengklasifikasikan *toxic comment* sudah dilakukan, baik dalam Bahasa Indonesia maupun Bahasa Asing. Dalam penelitian yang dilakukan oleh Saia dkk. menggunakan *Word Embeddings* sebagai *feature extraction*. Penelitian tersebut menggunakan 2 dataset yang sama dengan metode yang berbeda. Metode pertama menggunakan *Logistic Regression* sebagai *classifier* dan metode kedua menggunakan *K-Cross Validation* dengan nilai k = 10. Hasil dari penelitian tersebut menunjukkan bahwa menggunakan *Word Embeddings* sebagai *feature extraction* dapat meningkatkan akurasi dari sebuah *classifier* klasifikasi teks [16]. Terdapat juga penelitian yang menggunakan metode-metode *deep learning* dan menggunakan *data augmentation* karena adanya ketidakseimbangan antar kelas. Metode-metode yang digunakan sebagai *classifier* adalah *Convusional Neural Network, CNN Ensemble*, *Bidirectional LSTM* dan *Bidirectional GRU*. Hasil yang ditunjukkan dari penelitian tersebut adalah bahwa metode *CNN Ensemble* memiliki nilai *f-score* yang lebih tinggi dibandingkan dengan metode-metode lainnya[17].

III. METODE

A. Perancangan sistem

1. Sistem yang Dibangun

Penelitian ini membangun sistem klasifikasi teks toxic comment menggunakan metode Support Vector Machine (SVM) sebagai classifier, Information Gain sebagai feature selection dan TF-IDF sebagai feature extraction. Berikut adalah gambar rancangan sistem yang ingin dibangun.



GAMBAR 1.
Rancangan Sistem

2. Dataset

Dataset yang digunakan di-crawl menggunakan Python. Dataset diambil dari beberapa post pengguna twitter dengan topik "Nadiem Makarim Hapus Frasa Agama". Dataset yang diambil merupakan cuitan-cuitan dari sosial media Twitter dengan jumlah 711 data. Dataset tersebut kemudian diklasifikasikan menjadi 2 label yaitu toxic dan non-toxic. Dataset dengan label toxic berjumlah 584 dan label non-toxic berjumlah 37. Untuk pembagian dataset akan digunakan sebanyak 80% data uji dan 20% data latih.

3. Pre-processing

Untuk tahap pre-processing akan terbagi lagi menjadi beberapa tahap yaitu:

a. Case folding

Tahap lower casing adalah tahap untuk mengubah semua huruf pada kalimat menjadi huruf kecil.

b. Cleansing

Tahap cleansing dilakukan untuk menghilangkan tanda baca pada kalimat seperti titik dan koma.

c. Tokenizing

Tahap tokenizing adalah tahap untuk memisahkan satu kalimat menjadi kata per kata.

d. Stemming

Tahap stemming adalah tahap untuk mengubah kata yang berimbuhan menjadi kata aslinya.

4. Feature Selection dengan Information Gain

Setelah melalui data pre-processing akan dilakukan feature selection menggunakan metode Information Gain. Pada Information Gain ini akan dilakukan reduksi entropy pada sebuah variabel dengan cara mengubah dataset. Lalu nilai entropy dari sebelum dan sesudah data diubah akan dibandingkan.

5. Feature Extraction dengan TF-IDF

Feature extraction akan dilakukan menggunakan metode TF-IDF. TF-IDF menghitung frekuensi sebuah kata lalu membandingkannya dengan proporsi kata pada seluruh dokumen. Kata yang muncul lebih banyak memiliki bobot yang kecil karena bukan menjadi pembeda yang baik. Persamaan TF-IDF dapat ditulis sebagai:

$$w_{i,j} = tH_{i,j} \times \log\left(\frac{N}{dH_j}\right) \quad (7)$$

Di mana *t*Hi,j adalah banyaknya kata-i pada dokumen ke-j, N adalah total dokumen dan *d*Hi adalah banyaknya dokumen yang mengandung kata ke-i.

6. Pengujian Sistem

Untuk pengujian sistem akan dilakukan dengan menghitung presisi, *recall* dan *f-measure* algoritma tersebut. Lalu setelah hasil presisi, *recall* dan *f-measure* didapatkan maka akan dibandingkan dengan metode teks klasifikasi yang berbeda, untuk eksperimennya dilakukan dengan metode *Multinomial Naive-Bayes*.

IV. HASIL DAN PEMBAHASAN

A. Evaluasi

Dalam penelitian ini terdapat 3 skenario pengujian yang dilakukan. Skenario pertama yaitu pengujian terhadap feature selection Information Gain untuk melihat pengaruh threshold terhadap hasil akurasi dan prediksi. Sedangkan scenario kedua yaitu membandingkan metode SVM dengan metode Multinomial Naive Bayes.

1. Analisis pengaruh threshold feature selection

Pada skenario pertama, penulis melakukan pengujian terhadap pengaruh fitur seleksi Information Gain dalam mencari akurasi terbaik dengan menggunakan metode preprocessing, fitur ekstraksi TFIDF, implementasi klasifikasi Support Vector Machine, dan akan dievaluasi menggunakan confusion Matrix. Pertama, dilakukan data preprocessing agar data cuitan yang diambil menjadi lebih bersih dan mudah diproses. Lalu dilakukan fitur seleksi Information Gain untuk memilih atribut-atribut yang memiliki pengaruh tinggi terhadap klasifikasi. Hasil dari Information Gain digunakan untuk data latih dan data uji. Penulis melakukan 9 kali pengujian dengan 9 angka threshold yang berbeda yaitu antara 0.1 sampai dengan 0.9. Berikut adalah hasil pengujian yang didapatkan.

TABEL 2.
Hasil akurasi metode SVM dengan Informasi Gain

Threshold	F-1 Score	
	Non Toxic	Toxic
0.1	53%	59%
0.2	50%	58%
0.3	49%	53%
0.4	51%	49%
0.5	56%	56%
0.6	49%	50%
0.7	49%	49%
0.8	49%	49%
0.9	49%	49%

Berdasarkan pengujian yang dilakukan, akurasi tertinggi pada masing-masing label adalah 56% dengan threshold 0,5. Penggunaan *feature selection* menyebabkan fluktuasi nilai pada F-1 *score* tetapi dengan hasil yang tidak berbanding jauh.

2. Analisis klasifikasi tanpa menggunakan feature selection

Pada skenario kedua, penulis melakukan klasifikasi teks tanpa menggunakan *feature selection Information Gain* untuk mengetahui perbandingan antara hasil yang didapatkan. Berikut adalah hasil yang didapatkan.

TABEL 2.
Hasil akurasi Metode SVM tanpa Information Gain

Threshold	F-1 Score	
	Non Toxic	Toxic
0.1	43%	56%
0.2	41%	53%
0.3	40%	53%
0.4	39%	50%
0.5	37%	55%
0.6	40%	53%
0.7	39%	50%
0.8	39%	50%
0.9	39%	50%
Tanpa Information Gain	47%	46%

Berdasarkan hasil yang didapatkan, klasifikasi komentar *toxic* tanpa menggunakan feature selection *Information Gain* memiliki hasil F-1 *score* paling rendah di antara hasil F-1 *score* lainnya dengan hasil 48% pada label *non-toxic* dan *toxic*.

3. Analisis perbandingan metode SVM dengan metode Multinomial Naive Bayes

Pada skenario kedua, penulis membandingkan metode *Support Vector Machine* dengan metode *Multinomial Naive Bayes*. Penulis melakukan perbandingan tersebut untuk mengetahui apakah ada kekurangan ataupun kelebihan dari metode *Support Vector Machine*. Berikut adalah hasil akurasi yang didapatkan menggunakan metode *Multinomial Naive Bayes*.

TABEL 4
Hasil akurasi metode Multinomial Naive Bayes

Threshold	F-1 Score	
	Non Toxic	Toxic
0.1	53%	59%
0.2	50%	58%
0.3	49%	53%
0.4	51%	49%
0.5	56%	56%
0.6	49%	50%
0.7	49%	49%
0.8	49%	49%
0.9	49%	49%
Tanpa Information Gain	48%	48%

Berdasarkan pengujian yang dilakukan, akurasi tertinggi label *non-toxic* adalah 47% tanpa menggunakan *Information*

Gain sedangkan akurasi tertinggi pada label *toxic* adalah 56% dengan threshold 0.1. Dengan membandingkan hasil F-1 *score* metode SVM dan *Multinomial Naive Bayes,* metode SVM memiliki rata-rata hasil yang lebih tinggi.

V. KESIMPULAN

A. Kesimpulan

Berdasarkan hasil pengujian dari beberapa skenario terhadap klasifikasi teks, metode SVM memiliki hasil yang lebih baik dibandingkan dengan metode Multinomial Naive Bayes. Lalu dengan adanya *feature selection* dari *Information Gain* semakin memperkuat hasil akurasi dari metode tersebut. Information Gain menilai atribut-atribut dari setiap data dan mencari atribut yang paling tinggi sehingga kemunculan suatu kata akan dinilai.

REFERENSI

[1] B. B. BY BY Andrew Perrin, "Senior Communications Manager 202.419.4372 www.pewresearch.org RECOMMENDED CITATION: Andrew Perrin," 2015. [Online]. Available: www.pewresearch.org/internet

[2] S. Kemp, "Indonesian Digital Report 2020."

[3] J. W. Patchin and S. Hinduja, "Bullies Move Beyond the Schoolyard: A Preliminary Look at Cyberbullying," *Youth Violence Juv Justice*, vol. 4, no. 2, pp. 148–169, 2006, doi: 10.1177/1541204006286288.

[4] E. Whittaker and R. M. Kowalski, "Cyberbullying Via Social Media," *J Sch Violence*, vol. 14, no. 1, pp. 11–29, Jan. 2015, doi: 10.1080/15388220.2014.949377.

[5] Kaggle, "Toxic Comment Classification Challenge."

[6] T. Pranckevičius and V. Marcinkevičius, "Comparison of Naive Bayes, Random Forest, Decision Tree, Support Vector Machines, and Logistic Regression Classifiers for Text Reviews Classification," *Baltic Journal of Modern Computing*, vol. 5, no. 2, 2017, doi: 10.22364/bjmc.2017.5.2.05.

[7] S. Hassan, *2011 14th International Multitopic Conference.* IEEE, 2011.

[8] G. Tsoumakas, "Multi-Label Classification." [Online]. Available: http://www.dmoz.org/

[9] B. Yu, "An evaluation of text classification methods for literary study," in *Literary and Linguistic Computing*, 2008, vol. 23, no. 3, pp. 327–343. doi: 10.1093/llc/fqn015.

[10] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: A review," *Multimed Tools Appl*, vol. 78, no. 3, pp. 3797–3816, Feb. 2019, doi: 10.1007/s11042-018-6083-5.

[11] H. Uğuz, "A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm," *Knowl Based Syst*, vol. 24, no. 7, pp. 1024–1032, Oct. 2011, doi: 10.1016/j.knosys.2011.04.014.

[12] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers and Electrical Engineering*, vol. 40, no. 1, pp. 16–28, Jan. 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[13] D. Elreedy and A. F. Atiya, "A Comprehensive Analysis of Synthetic Minority Oversampling Technique (SMOTE) for handling class imbalance," *Inf Sci (N Y)*, vol. 505, pp. 32–64, Dec. 2019, doi: 10.1016/j.ins.2019.07.070.

[14] G. I. Webb, "Naïve Bayes," in *Encyclopedia of Machine Learning and Data Mining*, Springer US, 2016, pp. 1–2. doi: 10.1007/978-1-4899-7502-7_581-1.

[15] R. Susmaga, "Confusion Matrix Visualization."

[16] A. W. Romadon and D. Richasdy, "Analyzing TF-IDF and Word Embedding for Implementing Automation in Job Interview Grading," 2020.

M. Ibrahim, M. Torki, and N. El-Makky, "Imbalanced Toxic Comments Classification Using Data Augmentation and Deep Learning," in *Proceedings - 17th IEEE International Conference on Machine Learning and Applications, ICMLA 2018*, Jan. 2019, pp. 875–878. doi: 10.1109/ICMLA.2018.00141.