KLASIFIKASI KOMENTAR SPAM PADA YOUTUBE MENGGUNAKAN METODE NAÏVE BAYES, SUPPORT VECTOR MACHINE, DAN K-NEAREST NEIGHBORS

Burhanudin[1)], Yuniarti Musa'adah[2)], dan Yaya Wihardi[3)]
Departemen Pendidikan Ilmu Komputer, FPMIPA,UPI
Jl. Dr. Setiabudhi No. 229 Bandung
e-mail: beurhn@student.upi.edu[1)], yuniartimusaadah22@student.upi.edu[2)], yayawihardi@upi.edu[3)]

ABSTRAK

Situs jejaring sosial menjadi hal yang popular di zaman ini. Membagikan momen keseharian di media sosial sudah menjadi rutinitas harian. Orang-orang pun bisa lebih leluasa mengobrol mengenai objek yang dibagikan pada kolom komentar yang ada. Contohnya adalah komentar pada video Youtube. Namun kepopuleran media sosial membawa masalah dengan menarik para pengguna yang menyebarkan konten spam pada komentar. Dalam penelitian ini, akan dibahas mengenai klasifikasi komentar spam pada Youtube dengan beberapa metode yang diujikan. Dataset dengan jumlah 1956 data digunakan untuk training data. Hasil evaluasi model menggunakan cross validation menghasilkan metode Support Vector Machine dengan pendekatan Linear memiliki akurasi paling tinggi sebesar 91,92%. Diharapkan dengan penelitian ini bisa memberikan solusi sebagai upaya menangkal konten spam pada kolom komentar media sosial.

***Kata Kunci**: naive bayes, svm, knn.*

ABSTRACT

Social mediabecome popular in this day. Sharing the daily moments in social media has become a daily routine. People can also discuss about the post in the existing comment field. For example a comment on Youtube video. But the popularity of social media bring some problems with attracting users who spread spam content on comments. In this research, will be discussed about the classification of spam comments on Youtube with several methods tested. The dataset contains 1956 data,that used to train data. The result of model evaluation using cross validation resulted Support Vector Machine method with Linear approach has highest accuracy equal to 91,92%. Expectedby this research can provide solutions as an effort to prevent spam content in social media comment field..

***Keywords**: knn, naive bayes, svm*

I. PENDAHULUAN

Dalam keseharian, banyak masyarakat yang menggunakan media sosial untuk berbagai kepentingan, seperti berkomunikasi, berbagi momen sehari-hari, hingga berdagang. Media yang digunakan juga berbeda-beda, seperti menggunakan tulisan (status dan *tweet*), foto, maupun video. Orang-orang yang mngikuti pengguna media sosial (*friend, followers, subscriber*) pun bisa memberikan komentar terkait hal yang dibagikan tersebut. Namun hal ini terkadang dimanfaatkan oleh beberapa pihak untuk mengambil keuntungan atau setidaknya merugikan orang lain dengan komentar yang tidak relevan dengan objek yang dibagikan atau di-posting. Contohnya adalah komentar yang berisi tautan atau *link* yang mencurigakan, Komentar tersebut bisa berbahaya karena bisa saja tautan tersebut menyebarkan hal-hal yang tidak diinginkan hingga menyebabkan kerugian bagi penggunanya seperti kerusakan perangkat atau penyalahgunaan data pribadi yang didapatkan. Selain itu, contoh lainnya adalah *buzzer* yang membagikan komentar tidak relevan dan terus-menerus dengan tujuan lain. Hal ini dapat mengganggu kenyamanan pengguna dalam menggunakan media sosial.

Untuk mengatasi masalah tersebut diperlukan upaya untuk menyaring konten-konten yang dibagikan secara otomatis. Beberapa penelitian sudah dilakukan dalam penanganan terkait *spam*. Yu & Xu [1] melakukan penelitian untuk menguji beberapa metode dengan pendekatan *machine learning* untuk mem-*filterspam* pada *email*. Metode yang digunakan yaitu Naive Bayes, Neural Network, Support Vector Machine, dan Relevance Vector Machine. Hasilnya adalah SVM dan RVM memiliki akurasi paling tinggi diatas 90%. Penelitian lainnya dilakukan oleh McCord & Chuah [2] untuk mengklasifikasi *spam* pada media sosial Twitter. Metode yang digunakan yaitu Random Forest, Naive Bayes, Support Vector Machine, dan K-Nearest Neighbors. Hasilnya adalah Random Forest memiliki akurasi yang paling tinggi dengan persentase di atas 90% dengan berbagai fitur yang digunakan.

Telah banyak penelitian-penelitian lain terkait klasifikasi spam dengan berbagai metode yang ada. Dari dua kasus tersebut, penggunaan metode yang berbeda membuat hasil yang berbeda pula. Oleh karena itu, penelitian mengenai

perbandingan metode klasifikasi akan terus dilakukan pada berbagai kasus untuk mencari metode yang optimal pada setiap kasus. Pada artikel ini akan dilakukan perbandingan metode klasifikasi dengan kasus mendeteksi atau menyaring*spam* pada komentar di Youtube. Metode yang digunakan adalah Naive Bayes, Support Vector Machine, dan K-Nearest Neighbors dengan beberapa pendekatan yang ada berdasarkan *library* yang terdapat pada scikit-learn.org. Pendekatan yang ada yaitu pada Naive Bayes dengan Gaussian NB, Multinomial NB, dan Bernoulli NB. Kemudian SVM dengan pendekatan SVC, NuSVC, dan Linear SVC. Sedangkan untuk K-Nearest Neighbors dilakukan atau digunakan K yang berbeda, yaitu K = 1, K = 3, dan K = 5..

II. METODE

A. Naïve Bayes

Naïve Bayes adalah metode klasifikasi (*supervised learning*) dengan pendekatan probabilistik. Pendekatan ini membuat asumsi mengenai bagaimana data bisa dihasilkan dengan menempatkan model probabilistik untuk mewujudkannya. Klasifikasi Naïve Bayes merupakan metode yang sederhana dengan mangasumsikan bahwa semua atribut pada data tidak bergantung satu sama lain berdasarkan konteks kelas[3]. Rumus penggunaan Naïve Bayes pada klasifikasi teks adalah sebagai berikut.

$$P(C_i|X) = \frac{P(C_i)\times P(X|C_i)}{P(X)} \quad (1)$$

Dimana P merupakan probabilitas, X adalah atribut pada data, dan C adalah klasifikasi yang akan diprediksi, yaitu *spam* atau bukan *spam*. P(Ci) adalah probabilitas dari kelas, P(X|Ci) adalaha probabilitas dari data untuk kategori i, dan P(X) adalah probabilitas dari data [1].

Pada scikit-learn.org, Naïve Bayes memiliki tiga pendekatan, yaitu menggunakan Gaussian, Multinomial, dan Bernoulli. Gaussian Naïve Bayes menggunakan mean (μ) dan standar deviasi (σ) dengan rumus sebagai berikut [4].

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}}\exp(-\frac{\left(x_i-\mu_y\right)^2}{2\sigma_y^2}) \quad (2)$$

Kemudian untuk Multinomial Naïve Bayes menggunakan rumus berikut.

$$\theta_{yi} = \frac{N_{yi}+\alpha}{N_y+\alpha n} \quad (3)$$

Dimanaθ_{yi} adalah probabilitas $P(x_i|y)$ dari fitur i yang muncul dalam sample kelas y. $N_{yi} = \sum x \in T^{x_i}$ adalah jumlah fitur i yang muncul dalam sampel kelas y pada *training* set T. $N_y = \sum_{i=1}^{|T|} N_{yi}$ adalah jumlah total semua fitur untuk kelas y. Dan simbol α adalah *smoothing* untuk mencegah hasil probabilitas nol.

Untuk Bernoulli Naïve Bayes menggunakan rumus berikut,

$$P(x_i|y) = P(i|y)x_i + (1-P(i|y))(1-x_i) \quad (4)$$

Yang membedakan Bernoulli dengan aturan Multinomial adalah secara eksplisit memperhitungkan tidak terjadinya fitur i yang merupakan indikator untuk kelas y, di mana varian multinomial akan mengabaikan fitur yang tidak terjadi.

B. Support Vector Machine

Support Vector Machine adalah metode dengan model *supervised learning* yang digunakan untuk mengklasifikasi dan regresi dari suatu *dataset*. Model SVM adalah representasi dari data sebagai titik dalam ruang vektor, dipetakan hingga data dibagi dengan *space* yang jelas selebar mungkin. Kemudian data baru dipetakan ke dalam ruang yang sama untuk diprediksi berdasarkan di titik mana data baru tersebut dipetakan [5].

Dalam scikit-learn.org, SVM memiliki tiga pendekatan yaitu C-Support Vector Machine (SVC), Nu-Support Vector Machine (NuSVC), dan Linear Support Vector Machine (LinearSVC). SVC memiliki kompleksitas waktu yang lebih lama (*quadratic*) dan sulit diproses jika *dataset* yang digunakan lebih dari 10000 data. NuSVC hampir mirip dengan SVC namun menggunakan parameter untuk mengontrol jumlah *support vector*. Sedangkan LinearSVC adalah SVC dengan parameter kernel linear, memiliki lebih banyak fleksibilitias, dan dapat diproses

dengan baik untuk data dengan jumlah yang besar.

C. K-Nearest Neighbors

K-Nearest Neighbors atau KNN adalah metode untuk mengklasifikasikan objek atau data yang diuji ke dalam kelas di mana objek berada dan mana yang paling dekat dengan objek uji. Jika K lebih dari 1, anggota terdekat dari *learning set* dipilih dan objek yang diuji diklasifikasikan ke dalam kelas mayoritas dengan sistem *voting* [6].

Metode KNN terkadang kurang efektif digunakan dibeberapa kasus. Contohnya bila K yang digunakan berjumlah genap dan kelas dibagi menjadi dua. Sistem *voting* bisa tidak dapat dilakukan jika hasil yang didapat sama banyak atau seimbang.

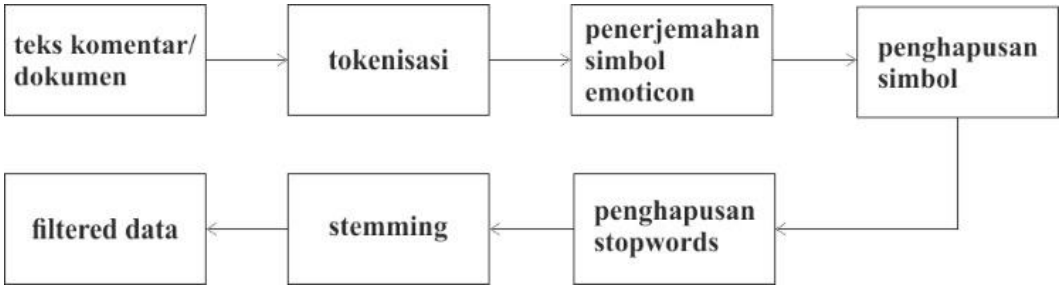
III. HASIL

Dataset yang digunakan dalam penelitian ini adalah 'Youtube Spam Collection Data Set' [7], berisi komentar berbahasa Inggris yang didapatkan dari 'Center for Machine Learning and Intelligent Systems' milik University of California. *Dataset* ini bersifat publik dan memiliki lima set data yang dikumpulkan dari 10 video dengan penonton paling banyak selama rentang waktu yang ditentukan. Total data komentar sebanyak 1956 data, lengkap dengan labelnya yaitu '0' untuk *non-spam* dan '1' untuk *spam*. Terdapat 1005 komentar *spam* dan 951 komentar *non-spam*.

Software yang digunakan untuk melakukan eksperimen adalah Python 3.5 (64 bit) dan perangkat yang digunakan adalah sebuah *notebook* dengan spesifikasi sebagai berikut: *Memory* 6 GB RAM, *Processor* Intel® Celeron® 1019Y (1.0GHz), dan *Hardisk* 500 GB

A. Preprocessing

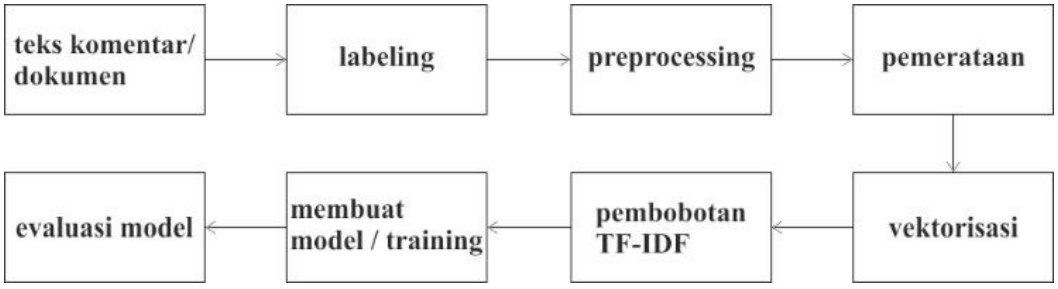
Sebelum dilakukan klasifikasi, data dipraproses terlebih dahulu. Praproses ini bertujuan untuk menghilangkan *noise* seperti simbol atau kata yang tidak diperlukan yang bisa memperlambat pemrosesan saat klasifikasi. Praproses yang dilakukan antara lain tokenisasi, penerjemahan simbol, *removing punctuation*, eliminasi *stopwords*, dan *stemming*.



Gambar 1. Alur Preprocessing

Alur praproses pada gambar 1 diawali dengan melakukan tokenisasi. Tokenisasi adalah mengidentifikasi kata kunci yang bermakna yang disebut token. Tokenisasi ini membagi kalimat menjadi kata-kata [8]. Kemudian penerjemahan simbol dilakukan untuk menerjemahkan simbol-simbol penting yang memiliki makna seperti *emoticon*. Contohnya symbol ':)' memiliki arti senang atau bahagia. *Removing punctuation* atau penghapusan simbol adalah menghilangkan karakter selain huruf pada suatu kata. Hal ini dilakukan untuk mengurangi *noise*[9]. Eliminasi *stopwords* adalah menghapus kata yang tidak diperlukan agar bisa mengurangi kompleksitas. Kata ganti, kata keterangan, preposisi, dan lain-lain yang digunakan secara terus-menerus di seluruh dokumen harus dihilangkan [8]. *Stemming* adalah mengubah setiap kata menjadi kata dasarnya dengan menghapus imbuhan [10]. Contohnya kata '*learning*' dengan kata '*learn*' akan disamakan nilainya. *Stemming* dalam penelitian ini menggunakan Porter Stemmer.

B. Training dan Evaluasi

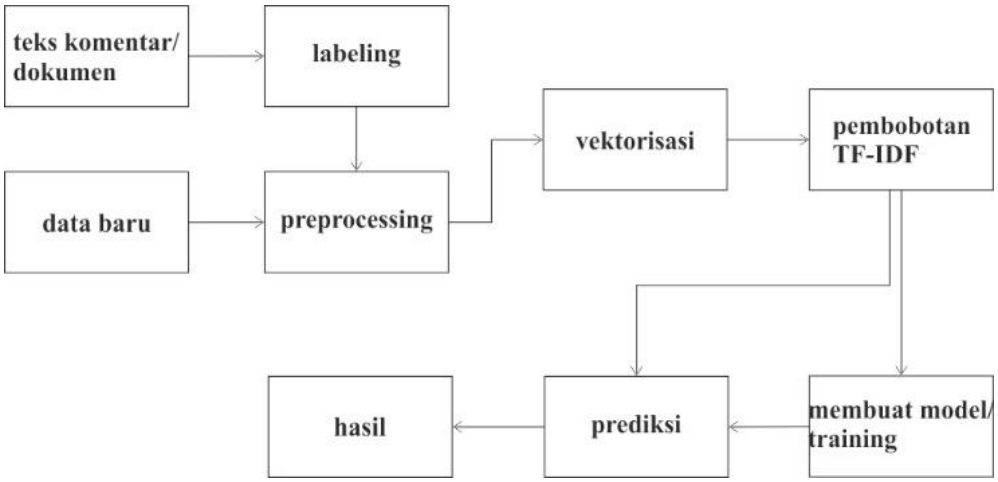


Gambar 2. Alur Pembuatan Model

Setelah data dipraproses, selanjutnya dilakukan pemerataan dengan proporsi penyebaran data dengan label *spam* dan *non-spam* yang seimbang. Hal ini bertujuan agar ketika model dievaluasi menggunakan *cross validation* dengan pembagian menjadi 5 bagian, penyebaran label data yang diuji akan seimbang sehingga hasilnya bisa lebih optimal.

Kemudian dilakukan vektorisasi berdasarkan *bag of words* pada setiap data untuk membangun fitur atau atribut secara keseluruhan. Selanjutnya pembobotan *bag of words* atau *Term Frequency* (TF) dikonversi menjadi bentuk *Term Frequency Inverse Document Frequency* (TF-IDF). TF-IDF menghitung nilai untuk setiap kata dalam dokumen melalui proporsi terbalik dari frekuensi kata dalam dokumen tertentu untuk persentase dokumen kata muncul. TF-IDF bisa mengkategorikan kata-kata yang relevan secara efisien yang dapat meningkatkan pengambilan *query*[11].

Setelah didapatkan atribut berupa nilai TF-IDF di setiap datanya, proses klasifikasi bisa dilakukan. Pada tahap pembuatan model atau *training data* ini, prosesnya menggunakan *library* GaussianNB, MultinomialNB, dan BernoulliNB untuk Naïve Bayes. Kemudian SVC, NuSVC, dan LinearSVC untuk SVM. Yang terakhir menggunakan KNeighborsClassifier untuk metode K-Nearest Neighbors dengan parameter n_neighbors masing-masing 1, 3, dan 5. Selanjutnya dilakukan evaluasi model menggunakan *cross validation* dengan 5 pembagian. Hasil yang didapat berupa nilai *precision*, *recall*, *f1-scores*, akurasi, dan waktu *training* pada masing-masing metode.



Gambar 3. Alur Testing data baru

Selain mengevaluasi model, dibangun juga program untuk testing data baru berdasarkan masukan dari *user* yang tidak terdapat dalam *dataset*. Alurnya adalah data baru dipraproses terlebih dahulu, kemudian dilakukan vektorisasi dan pembobotan TF-IDF yang menyesuaikan dengan *dataset*. Penyesuaian ini dilakukan karena untuk menambah atribut baru jika ada kata baru yang tidak terdapat dalam *dataset*. Selanjutnya dilakukan prediksi dengan model yang sudah dibangun dari *dataset*. Hasilnya adalah informasi terkait masukan dari *user*, apakah masukan tersebut termasuk ke dalam komentar *spam* atau bukan.

IV. PEMBAHASAN

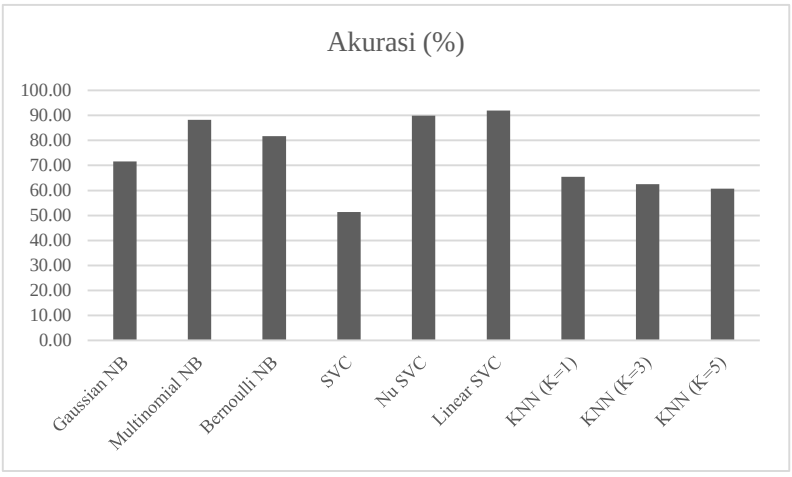
Berdasarkan evaluasi terhadap model menggunakan *cross validation*, hasil yang didapatkan adalah sebagai berikut,

TABEL I. HASIL EVALUASI MODEL MENGGUNAKAN*CROSS VALIDATION*

Metode		Precision	Recall	F1-Scores	Akurasi (%)	Waktu (detik)
Gaussian NB		0.73	0.72	0.71	71.63	4.67
Multinomial NB		0.89	0.88	0.88	88.24	1.69
Bernoulli NB		0.86	0.82	0.81	81.75	3.44
SVC		0.26	0.50	0.34	51.38	272.72
Nu SVC		0.91	0.90	0.90	89.98	177.36

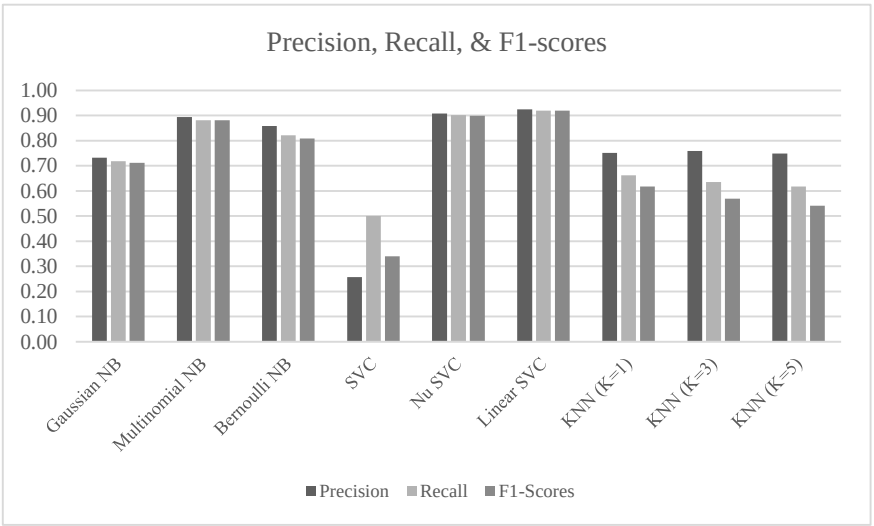
Linear SVC		0.92	0.92	0.92	91.92	1.47
KNN (K=1)		0.75	0.66	0.62	65.39	56.00
KNN (K=3)		0.76	0.63	0.57	62.53	116.31
KNN (K=5)		0.75	0.62	0.54	60.69	124.28

Hasil *cross validation* menunjukkan metode Support Vector Machine dengan pendekatan linear memiliki akurasi paling tinggi sebesar 91.92%. Waktu proses paling singkat pun dipegang oleh LinearSVC dengan 1,47 detik. Metode Multinomial NB pun memiliki hasil yang cukup baik dengan akurasi sebesar 88.24% dengan waktu hanya 1.69 detik. Begitupun dengan Nu SVC yang memiliki akurasi hampir 90% namun dengan waktu yang cukup lama, yaitu hampir 3 menit untuk membuat modelnya. Hal ini berbanding terbalik dengan SVM menggunakan SVC yang memiliki akurasi paling rendah sebesar 51.38% dengan waktu tempuh yang sangat lama, yaitu 272.72 detik.



Gambar 4. Tingkat akurasi setiap metode

Dari gambar 4 terlihat LinearSVC yang memiliki tingkat akurasi paling tinggi, diikuti NuSVC dan Multinomial Naïve Bayes dengan akurasi di atas 85%. Artinya ketiga metode ini sangat baik digunakan untuk mengklasifikasi komentar *spam* berjenis teks ini. Hasil lainnya adalah metode K-Nearest Neighbors memiliki tingkat akurasi yang stabil namun cenderung menurun seiring jumlah K yang semakin besar. Hal ini kemungkinan disebabkan oleh jumlah data yang terambil sebanyak K memiliki gangguan atau *error* dari data yang salah.



Gambar 5. Tingkat *precision, recall, dan f1-scores*setiapmetode

Hasil darinilai*precision*, *recall*, dan *f1-scores*darimetode yang diujiterlihatdarigambar 5 bahwametode Multinomial Naïve Bayes, Bernoulli Naïve Bayes, Nu SVC, dan Linear SVC memilikitingkat*precision*, *recall*, dan *f1-scores* paling tinggidibandingmetodelainnya.

V. SIMPULAN DAN SARAN

Berdasarkan penelitian yang telah dilakukan, diambil kesimpulan bahwa tidak semua algoritma atau metode klasifikasi bisa mengklasifikasikan data berbentuk teks dengan baik. Pada penelitian terkait yang dilakukan oleh Yu & Xu [1] menyebutkan bahwa metode Support Vector Machine memiliki akurasi yang paling baik dibanding metode lainnya. Namun tidak semua pendekatan dalam SVM memiliki hasil yang baik, seperti menggunakan *library* SVC pada scikit-learn yang memiliki akurasi hanya sekitar 50% dengan waktu *training* yang sangat lama. Namun dari tiga metode dengan masing-masing tiga pendekatan yang diujikan untuk mengklasifikasi komentar *spam* pada Youtube, metode Support Vector Machine dengan pendekatan Linear (LinearSVC) memiliki hasil yang cukup baik.

Hasil yang berbeda bisa didapatkan pada kasus yang lain. Sehingga diharapkan penelitian ini bisa terus berlanjut dengan kasus-kasus lain khususnya dalam pemrosesan teks. Pemilihan fitur atau atribut pun berpengaruh kepada efisiensi dan efektifitas dari metode yang digunakan. Diharapkan juga praproses yang dilakukan pada penelitian yang lain bisa lebih dioptimalkan lagi agar mendapat data yang lebih murni dan lebih berbobot. Kekurangan dari penelitian ini adalah masih banyak kata yang tidak berubah ke bentuk aslinya karena penulisan komentar oleh pengguna yang sangat bebas, seperti kesalahan penulisan, teks yang tidak lengkap, ataupun *slang* yang digunakan.

REFERENSI

[1] B. Yu and Z.-b. Xu, "A comparative study for content-based dynamic spam classification using four machine learning algorithms," *Knowledge-Based Sybermatika*, vol. 21, no. 4, pp. 355-362, 2008.

[2] M. McCord and M. Chuah, "Spam Detection on Twitter Using Traditional Classifiers," in *international conference on Autonomic and trusted computing*, Berlin, 2011.

[3] A. McCallum and K. Nigam, "A Comparison of Event Models for Naive Bayes Text Classification," in *AAAI-98 workshop on learning for text categorization*, 1998.

[4] V. Metsis, I. Androutsopoulos and G. Paliouras, "Spam Filtering with Naive Bayes – Which Naive Bayes?," in *3rd Conf. on Email and Anti-Spam (CEAS)*, 2006.

[5] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.

[6] D. Coomans and D. L. Massart, "Alternative k-nearest neighbour rules in supervised pattern recognition : Part 1. k-Nearest neighbour classification by using alternative voting rules," *Analytica Chimica Acta*, vol. 136, pp. 15-27, 1982.

[7] T. C. Alberto, J. V. Lochter and T. A. Almeida, "TubeSpam: Comment Spam Filtering on YouTube," in *International Conference on Machine Learning and Applications*, Miami, 2015.

[8] R. C. Balabantaray, C. Sarma and M. Jha, "Document Clustering using K-Means and K-Medoids," *International Journal of Knowledge Based Computer System*, vol. 1, no. 1, pp. 7-13, 2013.

[9] E. Rasywir and A. Purwarianti, "Eksperimen pada Sistem Klasifikasi Berita Hoax Berbahasa Indonesia Berbasis Pembelajaran Mesin," *Jurnal Cybermatika*, vol. 3, no. 2, pp. 1-8, 2015.

[10] M. A. Fauzi, A. Z. Arifin, S. C. Gosaria and I. S. Prabowo, "Indonesian News Classification Using Naïve Bayes and Two-Phase Feature Selection Model," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 8, no. 3, pp. 610-615, 2017.

[11] J. Ramos, "Using TF-IDF to Determine Word Relevance in Document Queries," in *Instructional Conference On Machine Learning*, 2003.