

Analisis Sentimen pada Ulasan Film dengan Kombinasi Seleksi Fitur Chi-Square dan TF-IDF menggunakan Metode KNN

Rizki Ramadhan Al-Mubaraq¹, Said Al Faraby², Mahendra Dwifabri Purbolaksono³

^{1,2,3} Universitas Telkom, Bandung

¹rizkira@students.telkomuniversity.ac.id, ²saidalfaraby@telkomuniversity.ac.id,

³mahendradp@telkomuniversity.ac.id

Abstrak

Terdapat banyak film yang dapat menjadi pilihan untuk ditonton, namun belum tentu film-film tersebut memiliki kualitas yang bagus. Ulasan film menjadi sangat penting dikarenakan mampu memberikan informasi mengenai kualitas suatu film sehingga dapat dijadikan referensi untuk menonton suatu film. Dikarenakan data mengenai ulasan film berjumlah sangat banyak dan membutuhkan waktu lama jika membacanya secara manual satu per satu, diperlukan sistem yang dapat melakukan analisis sentimen dalam waktu yang singkat. Metode *machine learning* seperti *K-Nearest Neighbor* (KNN) dapat digunakan untuk melakukan analisis sentimen, namun memiliki performansi yang kurang baik jika fitur yang digunakan tidak ditentukan dengan baik. Untuk menentukan fitur yang akan digunakan, penggunaan ekstraksi fitur dan seleksi fitur perlu digunakan. Dalam penelitian ini, ekstraksi fitur yang digunakan adalah *Term Frequency-Inverse Document Frequency* (TF-IDF) dan seleksi fitur yang digunakan adalah *chi-square*. Dengan kombinasi antara metode *machine learning* KNN, ekstraksi fitur TF-IDF, dan seleksi fitur *chi-square*, dapat memberikan hasil evaluasi performansi terbaik pada $K = 37$ dengan nilai akurasi akurasi 80.75%, *precision* 83.24%, *recall* 77.00%, dan *F1-measure* 80.00%.

Kata kunci : analisis sentimen, TF-IDF, *chi-square*, KNN

Abstract

Many films can be an option to watch, but not necessarily films that are of good quality. Film reviews are very important because they are able to provide information about the quality of a film so that it can be used as a reference for watching a film. Due to the large amount of data regarding film reviews and it takes a long time to read them manually one by one, a system that can perform sentiment analysis in a short time is needed. Machine learning methods such as *K-Nearest Neighbor* (KNN) can be used to perform sentiment analysis, but have poor performance if the features used are not well defined. To determine the features, the use of feature extraction and feature selection needs to be used. In this study, *Term Frequency-Inverse Document Frequency* (TF-IDF) is used for feature extraction and *chi-square* is used for feature selection. With a combination of KNN methods, TF-IDF feature extraction, and *chi-square* feature selection, it can provide the best performance evaluation results at $K = 37$ with an accuracy value of 80.75% accuracy, 83.24% precision, 77.00% recall, and *F1-measure* 80.00%.

Keywords: sentiment analysis, TF-IDF, *chi-square*, KNN

1. Pendahuluan

Latar Belakang

Menonton film merupakan suatu aktivitas yang dilakukan untuk mengisi rasa bosan atau menghilangkan stres yang dilakukan oleh seorang individu atau sekelompok orang. Ulasan mengenai suatu film sangat penting dikarenakan informasi tersebut dapat dijadikan referensi untuk menonton suatu film. Dikarenakan data mengenai ulasan film berjumlah sangat banyak, memerlukan banyak waktu untuk membaca ulasan tersebut satu per satu [1]. Berdasarkan kondisi tersebut, analisis sentimen dalam ulasan film merupakan topik yang menarik untuk diselesaikan dengan menggunakan *machine learning*, karena secara otomatis akan melakukan proses klasifikasi dan mempersingkat waktu.

Terdapat berbagai metode *machine learning* yang digunakan untuk melakukan proses klasifikasi, salah satunya menggunakan metode *K-Nearest Neighbor* (KNN). Metode KNN memiliki beberapa keunggulan jika dibandingkan dengan metode lainnya seperti efektif terhadap data latih yang ukurannya besar dan memiliki banyak *noise*, tetapi metode KNN memiliki performa hasil klasifikasi yang kurang baik apabila fitur yang digunakan tidak ditentukan dengan baik [2]. Untuk menentukan fitur yang akan digunakan, penggunaan seleksi fitur dan ekstraksi fitur perlu dilakukan.

Dalam penelitian ini, metode *machine learning* yang digunakan adalah KNN dengan menggunakan seleksi fitur *chi-square* dan ekstraksi fitur *Term Frequency-Inverse Document Frequency* (TF-IDF). Berdasarkan

penelitian yang dilakukan oleh Luqman Rizky Dharmawan [3], penggunaan seleksi fitur *chi-square* dan metode KNN pada dataset yang diambil dari twitter tentang layanan Sistem Informasi Akademik Mahasiswa Universitas Brawijaya (SIAM UB) menghasilkan nilai akurasi sebesar 86%. Untuk ekstraksi fitur, TF-IDF dipilih dikarenakan mampu mengasosiasikan setiap kata dalam dokumen dalam bentuk angka yang mewakili seberapa relevan setiap kata tersebut dalam suatu dokumen [4].

Topik dan Batasannya

Topik pada penelitian tugas akhir ini adalah melakukan pengujian kombinasi antara seleksi fitur *chi-square*, ekstraksi fitur TF-IDF, dan metode KNN terhadap *dataset* ulasan film. Untuk batasannya, *dataset* yang digunakan merupakan ulasan film dalam Bahasa Inggris dari situs IMDb dengan jumlah data yang digunakan sebanyak 2001 data, terdiri dari 1000 data ulasan positif dan 1001 data ulasan negatif.

Tujuan

Tujuan yang ingin dicapai pada penelitian tugas akhir ini adalah menganalisis hasil performansi kombinasi ekstraksi fitur TF-IDF dan metode KNN terhadap penggunaan stemming, *threshold* seleksi fitur *chi-square* yang digunakan, dan pengaruh perubahan nilai K metode KNN pada *dataset* ulasan film dalam Bahasa Inggris.

Organisasi Tulisan

Pada bab 2 dibahas mengenai studi terkait penelitian yang dilakukan, bab 3 membahas teori dan perancangan sistem penelitian, bab 4 membahas evaluasi model, dan bab 5 membahas kesimpulan dari penelitian yang digunakan.

2. Studi Terkait

Dalam melakukan analisis sentimen, penggunaan metode *machine learning* sangat membantu dikarenakan dapat melakukan klasifikasi terhadap suatu data yang jumlahnya banyak dalam waktu yang singkat [1]. Dalam penelitian yang dilakukan oleh Novelty Octoviani Faomasi Daeli [5], penggunaan berbagai metode *machine learning* terhadap *dataset* ulasan film dalam Bahasa Inggris yang terdiri dari 1000 dokumen negatif dan 1000 dokumen positif menghasilkan nilai akurasi yang berbeda-beda. Berdasarkan penelitian tersebut, metode KNN memiliki nilai akurasi tertinggi jika dibandingkan dengan metode *Naïve Bayes* (NB), *Random Forest* (RF), dan *Support vector Machine* (SVM), yaitu sebesar 96.15%. Meskipun memiliki performa yang baik, metode KNN masih memiliki kekurangan, yaitu nilai K yang ditentukan sangat mempengaruhi hasil evaluasi performansi. Dalam penelitian yang dilakukan oleh Masoud Maleki [6], pencarian nilai K yang optimal perlu dilakukan agar memperoleh hasil evaluasi performansi yang optimal juga.

Selain pencarian nilai K, terdapat hal lain yang dapat meningkatkan hasil evaluasi performansi. Penggunaan seleksi fitur merupakan salah satu contoh untuk meningkatkan hasil evaluasi performansi [7]. Dalam penelitian yang dilakukan oleh Ukhti Ikhsani Larasati [8], penggunaan seleksi fitur *chi-square* yang sebelumnya dilakukan ekstraksi fitur terlebih dahulu dengan menggunakan TF-IDF pada metode SVM dengan menggunakan *dataset* ulasan film dalam Bahasa Inggris yang terdiri dari 500 data positif dan 500 data negatif mampu meningkatkan akurasi dari 68.8% menjadi 80.2%. Penelitian tersebut menunjukkan penggunaan seleksi fitur mampu meningkatkan akurasi sebesar 11.5%.

Penelitian berikutnya perihal pemilihan seleksi fitur yang dilakukan oleh H. M. Keerthi Kumar [9], yang menunjukkan penggunaan seleksi fitur berbeda belum tentu hasilnya juga akan berbeda. Dalam penelitian tersebut, dilakukan empat buah pengujian berupa kombinasi antara KNN dengan seleksi fitur *Information gain* (IG), *chi-square*, *correlation*, dan *Regularized Locality Preserving Indexing* (RLPI). Semua kombinasi tersebut dilakukan terhadap *dataset* yang diambil dari situs IMDb secara random sebanyak 5000 data. Hasil dari keempat pengujian tersebut, seleksi fitur *chi-square* memberikan hasil terbaik dengan akurasi 72.267% dan peningkatan *F1-measure* tertinggi, yaitu sebesar 21.72%. Hal tersebut membuktikan bahwa penggunaan seleksi fitur *chi-square* memiliki performa yang paling bagus jika dikombinasikan dengan KNN.

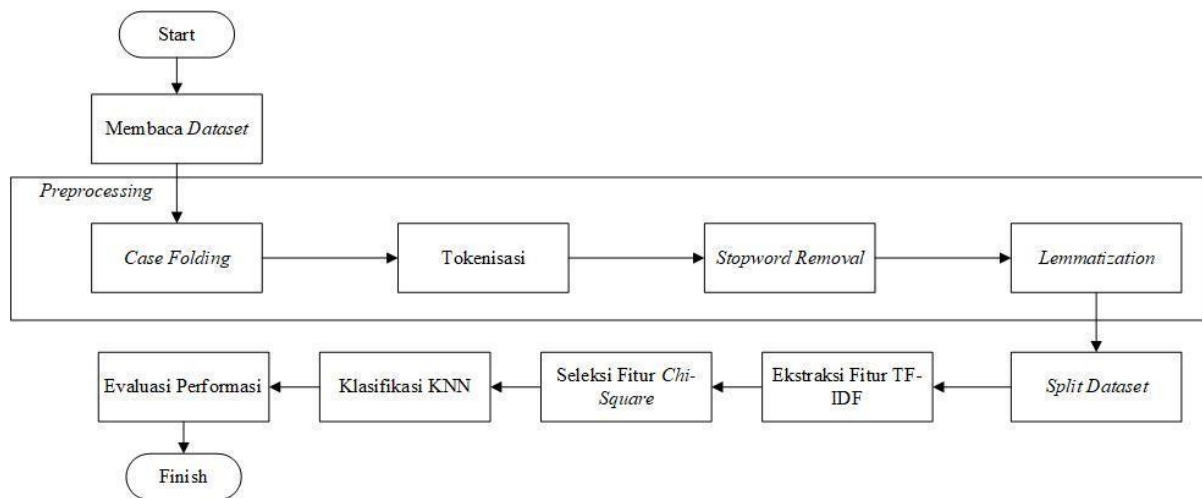
Penelitian berikutnya berkaitan dengan penggunaan [4] *dataset* yang berbeda mampu mempengaruhi performansi hasil klasifikasi. Berdasarkan penelitian yang dilakukan oleh Luqman Rizky Dharmawan [3], kombinasi antara ekstraksi fitur TF-IDF, seleksi fitur *chi-square*, dan KNN terhadap *dataset* yang diambil dari twitter tentang layanan Sistem Informasi Akademik Mahasiswa Universitas Brawijaya (SIAM UB) menghasilkan nilai akurasi sebesar 86%, sedangkan berdasarkan penelitian yang dilakukan oleh Indah Listiowarni [7], penggunaan ekstraksi fitur, seleksi fitur, dan metode *machine learning* yang sama terhadap *dataset* dari data soal ujian SMA menghasilkan nilai akurasi 79.36%. Dari kedua penelitian tersebut, dapat disimpulkan bahwa meskipun menggunakan kombinasi seleksi fitur, ekstraksi fitur, dan metode *machine learning* sama, hasilnya akan berbeda jika *dataset* yang digunakan berbeda.

Selain *dataset* yang digunakan, pada penelitian sebelumnya tidak menentukan batas frekuensi dokumen minimal. Berdasarkan penelitian yang dilakukan oleh Zahratul Fikrina [10], frekuensi dokumen minimal pada

proses TF-IDF memiliki pengaruh untuk melakukan klasifikasi agar hasilnya lebih baik. Oleh karena itu, penulis ingin membangun analisis sentimen terhadap *dataset* ulasan film dalam Bahasa Inggris dari situs IMDb dengan jumlah data yang digunakan sebanyak 2001 data dan klasifikasi hanya berupa ulasan film positif dan negatif menggunakan kombinasi ekstraksi fitur TF-IDF dengan menerapkan frekuensi dokumen minimal, seleksi fitur *chi-square*, dan metode *machine learning* KNN untuk mendapatkan hasil performa yang terbaik.

3. Sistem yang Dibangun

Dalam penelitian ini, analisis sentimen ulasan film dilakukan dengan menggunakan ekstraksi fitur TF-IDF, seleksi fitur *chi-square*, dan metode *machine learning* KNN. Perancangan sistem yang akan dibangun dalam penelitian ini dapat dilihat pada Gambar 1.



Gambar 1. Perancangan Sistem

Pada Gambar 1, langkah awal yang dilakukan adalah membaca *dataset* terlebih dahulu. Dalam penelitian kali ini, *dataset* yang digunakan adalah data ulasan film dalam Bahasa Inggris dari situs IMDb. Selanjutnya masuk ke tahap *preprocessing* yang terdiri dari *case folding*, tokenisasi, *stopword removal*, dan *lemmatization*. Setelah melakukan tahap *preprocessing*, tahap selanjutnya adalah melakukan ekstraksi fitur dengan menggunakan TF-IDF yang bertujuan untuk memberikan bobot pada setiap kata. Setelah melakukan ekstraksi fitur, langkah selanjutnya adalah melakukan reduksi fitur yang *irrelevant* dengan menggunakan seleksi fitur *chi-square*. Selanjutnya, klasifikasikan dengan metode KNN dan lakukan evaluasi performansi dengan hasil yang dinyatakan dalam bentuk akurasi, *precision*, *recall*, dan *F1-measure*.

3.1 Membaca *dataset*

Membaca *dataset* merupakan langkah awal yang dilakukan dalam penelitian ini. *Dataset* yang digunakan merupakan data ulasan film dalam Bahasa Inggris dari situs IMDb dengan jumlah data yang digunakan sebanyak 2001 data, terdiri dari 1001 data ulasan negatif dan 1000 data ulasan positif. Berikut adalah contoh dari *dataset* yang dapat dilihat pada Tabel 1.

Tabel 1. *Dataset* Penelitian

Ulasan	Label
"though it is by no means his best work, <i>laissez-passer</i> is a distinguished and distinctive effort by a bona-fide master , fascinating film replete with reward to be had by all willing to make the effort to reap them "	Positif
"A visually flashy but narratively opaque and emotionally vapid exercise in style and mystification . The story is also as unoriginal as they come , already having been recycled more times than i'd care to count (4 times) :(. "	Negatif

3.2 *Preprocessing*

Preprocessing merupakan langkah penting untuk meningkatkan efisiensi data [11]. Sebelum melakukan *preprocessing*, drop satu data ulasan negatif agar total *dataset* menjadi seimbang antara data ulasan positif dengan data ulasan negatif. Setelah melakukan hal tersebut, melanjutkan ke tahapan *preprocessing* yang dapat dilihat pada Gambar 2.



Gambar 2. Tahapan Preprocessing

Pada tahapan *case folding*, semua huruf akan diubah kedalam bentuk *lowercase*. Berikut adalah contoh dari penerapan tahapan *case folding* yang dapat dilihat pada Tabel 2.

Tabel 2. Penerapan Case Folding

Kalimat Masukkan	<i>“A visually flashy but narratively opaque and emotionally vapid exercise in style and mystification . The story is also as unoriginal as they come , already having been recycled more times than i’d care to count (4 times) :(. “</i>
Hasil	<i>“a visually flashy but narratively opaque and emotionally vapid exercise in style and mystification . the story is also as unoriginal as they come , already having been recycled more times than i’d care to count (4 times) :(. “</i>

Selanjutnya tahap tokenisasi yang merupakan tahapan untuk memisahkan kata per kata dalam suatu dokumen. Sebelum memisahkan kata per kata, bersihkan data terlebih dahulu dari simbol atau angka yang ada dalam dokumen, menghapus *whitespaces* pada dokumen, dan ubah *apostrophe/short word* ke bentuk asalnya, misalnya kata *“I’ll”* menjadi *“I will”*. Jika sudah melakukan ketiga hal tersebut, lakukan proses tokenisasi. Berikut adalah contoh penerapan tahapan tokenisasi yang menggunakan data hasil dari Tabel 2.

Tabel 3. Penerapan Tokenisasi

Hasil	<i>['visually', 'flashy', 'but', 'narratively', 'opaque', 'and', 'emotionally', 'vapid', 'exercise', 'in', 'style', 'and', 'mystification', 'the', 'story', 'is', 'also', 'as', 'unoriginal', 'as', 'they', 'come', 'already', 'having', 'been', 'recycled', 'more', 'times', 'than', 'would', 'care', 'to', 'count', 'times']</i>
--------------	--

Setelah melakukan tokenisasi, tahap selanjutnya adalah melakukan *stopword removal* yang bertujuan untuk menghapus kata-kata yang tidak mengandung konten. Berikut adalah hasil dari penerapan *stopword removal* yang menggunakan data hasil dari Tabel 3.

Tabel 4. Penerapan Stopword Removal

Hasil	<i>['visually', 'flashy', 'narratively', 'opaque', 'emotionally', 'vapid', 'exercise', 'style', 'mystification', 'story', 'also', 'unoriginal', 'come', 'already', 'recycled', 'times', 'would', 'care', 'count', 'time']</i>
--------------	---

Terakhir, tahap *lemmatization* yang merupakan tahapan untuk mengubah setiap kata menjadi kata dasar. Hasil dari penerapan tahap *lemmatization* dapat dilihat pada Tabel 5 yang menggunakan masukan dari hasil pada Tabel 4.

Tabel 5. Penerapan Lemmatization

Hasil	<i>['visually', 'flashy', 'narratively', 'opaque', 'emotionally', 'vapid', 'exercise', 'style', 'mystification', 'story', 'also', 'unoriginal', 'come', 'already', 'recycle', 'time', 'would', 'care', 'count', 'time']</i>
--------------	---

3.3 Split dataset

Split dataset merupakan proses untuk memisahkan antara data latih dengan data tes. *Dataset* hasil *preprocessing* akan dibagi menjadi 80% data latih dan 20 % data tes. Hasil dari *split dataset* dapat dilihat pada Tabel 6.

Tabel 6. Hasil *Split Dataset*

	Positif	Negatif
Data latih	800	800
Data tes	200	200

3.4 Ekstraksi fitur TF-IDF

Term Frequency-Inverse Document Frequency (TF-IDF) merupakan suatu metode algoritma yang digunakan untuk memberi bobot pada setiap kata yang umum digunakan sebagai strategi untuk mengklasifikasikan teks atau dokumen [12]. Pada tahapan TF-IDF, dokumen akan dipecah kedalam bentuk *unigram* atau satu kata. Proses pemberian bobot pada kata dengan TF-IDF terdiri dari proses menghitung *Term Frequency* (TF) dan *Inverse Document frequency* (IDF). Agar nilai TF tidak terlalu jauh perbedaannya, perlu dilakukan proses *scaling*. Berikut adalah persamaan TF yang menggunakan proses *scaling*.

$$W_{td} = \begin{cases} 1 + \log(TF_{td}) & \text{if } TF_{td} > 0 \\ 0 & \text{if } TF_{td} = 0 \end{cases} \quad (1)$$

TF_{td} menyatakan jumlah berapa banyak suatu kata dalam suatu dokumen. Setelah menemukan nilai W_{td} , langkah selanjutnya adalah mencari nilai IDF. Berikut adalah persamaan untuk mencari nilai IDF.

$$IDF_t = \log \left(\frac{1+n}{1+df_t} \right) + 1 \quad (2)$$

n menyatakan total jumlah dataset. df_t jumlah dokumen yang mengandung suatu kata. Setelah menemukan nilai IDF_t, langkah selanjutnya adalah mengalikan nilai W_{td} dengan IDF_t sehingga nilai TF-IDF dapat diperoleh. Berikut adalah persamaan TF-IDF.

$$TF\ IDF_t = W_{td} \times IDF_t \quad (3)$$

3.5 Seleksi fitur *chi-square*

Chi-square merupakan salah satu metode seleksi fitur yang termasuk metode filter untuk melakukan reduksi fitur yang irrelevant [7]. Berikut adalah persamaan yang digunakan untuk menerapkan seleksi fitur chi-square.

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i} \quad (4)$$

Variabel c merupakan *degrees of freedom* yang dapat dicari dengan cara total baris dikurang satu, O merupakan *observed value* yang merupakan jumlah kemunculan tiap nilai dalam suatu fitur terhadap tiap kelas, dan E merupakan *expected value* yang bisa dicari dengan cara menggunakan formula 5.

$$E_i = \frac{a \cdot b}{N} \quad (5)$$

Variable N merupakan jumlah data, a merupakan total jumlah kemunculan tiap nilai dalam suatu fitur, dan b merupakan total jumlah tiap kelas. Setelah nilai *chi-square* didapatkan, langkah selanjutnya adalah mencari *p-value* yang bisa dilakukan dengan cara mengecek Tabel *chi-square*. Tabel *chi-square* memuat berbagai macam informasi seperti *degree of freedom*, *p-value*, dan nilai *chi-square*. Contoh Tabel *chi-square* dapat dilihat pada Tabel 7. Untuk contoh penggunaannya, terdapat nilai *chi-square* 3.841 untuk suatu fitur dengan *degree of freedom* satu, setelah mengecek ke Tabel *chi-square*, nilai *p-value* untuk suatu fitur tersebut adalah 0.05. Untuk melakukan penyeleksian fitur, buang fitur yang memiliki nilai *p-value* lebih kecil dari tingkat signifikansi yang bisa dicari dengan menggunakan formula 6

$$\alpha = 1 - \text{Confident Level} \quad (6)$$

α menyatakan tingkat signifikansi dan *Confident Level* menyatakan tingkat kepercayaan terhadap *null hypothesis* atau menganggap semua fitur merupakan fitur yang *independent* [13]. Berdasarkan formula 6, tingkat signifikansi juga dapat diartikan kebalikan dari *Confident Level*.

Tabel 7. Tabel Chi-Square

Df	P						
	0.995	0.975	0.2	0.1	0.05	0.025	0.02
1	0.0000393	0.000982	1.642	2.706	3.841	5.024	5.412
2	0.0100	0.0506	3.219	4.605	5.991	7.378	7.824
3	0.0717	0.216	4.642	6.251	7.815	9.348	9.837
4	0.207	0.484	5.989	7.779	9.488	11.143	11.668

3.6 Klasifikasi KNN

K-Nearest Neighbor (KNN) merupakan suatu metode yang menggunakan algoritma *supervised learning* untuk mengklasifikasikan data berdasarkan nilai K tetangga terdekat [5]. KNN mengukur *similarity* dengan cara mencari jarak terdekat antara data yang akan dievaluasi dengan K tetangga terdekatnya dalam data latih. Salah satu metode pengukuran jarak yang dapat digunakan untuk menghitung *similarity* adalah *Euclidean Distance* dengan persamaan sebagai berikut.

$$d = \sqrt{(|x_2 - x_1|)^2 + (|y_2 - y_1|)^2} \quad (6)$$

3.7 Evaluasi performansi

Dalam mengukur performansi suatu metode, *F1-Measure* digunakan untuk melakukan suatu perhitungan evaluasi yang mengkombinasikan nilai antara *precision* dan *recall* [7]. Untuk hasil prediksi, akan dipresentasikan dalam tabel 9 dengan informasi sebagai berikut.

- *True Positive* (TP), nilai positif yang diprediksi dengan benar
- *False Negative* (FN), nilai positif yang dianggap nilai negatif
- *False Positive* (FP), nilai negatif yang dianggap nilai positif
- *True Negative* (TN), nilai negatif yang diprediksi dengan benar

Tabel 8. Confusion Matrix

	<i>Predicted positive</i>	<i>Predicted negative</i>
<i>Actual positive</i>	TP	FN
<i>Actual negative</i>	FP	TN

Terdapat beberapa parameter yang dapat digunakan dari pengolahan *confusion matrix* yang akan dijelaskan sebagai berikut.

- Akurasi

Akurasi merupakan tingkat kedekatan antara nilai prediksi dengan nilai aktual dengan persamaan sebagai berikut.

$$Akurasi = \frac{TP+TN}{TP+FP+FN+TN} \quad (7)$$

- *Precision*

Precision merupakan jumlah sampel positif diklasifikasikan benar dibagi total sampel positif. *Precision* menunjukkan tingkat ketepatan antara informasi yang diminta dengan jawaban yang diberikan sistem. Berikut adalah persamaan dari *Precision*.

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

- *Recall*

Recall merupakan tingkat keberhasilan sistem dalam menemukan informasi. Nilai *Recall* didapatkan melalui persamaan sebagai berikut.

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

- *F1-measure*

F1-measure merupakan nilai rata-rata harmonik dari *Precision* dan *Recall*. Berikut adalah persamaan *F1-measure*.

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

4. Evaluasi

Proses klasifikasi dengan menggunakan metode KNN dilakukan untuk mengetahui ulasan film dari situs IMDb yang bernilai positif atau negatif. Agar tercapai hasil optimal, dibangun skenario pengujian dalam menghasilkan performa klasifikasi yang akan dijadikan perbandingan. Skenario pengujian dapat dilihat pada Tabel 9.

Tabel 9. Skenario Pengujian

No	Skenario	Tujuan
1	Membandingkan penetapan frekuensi dokumen minimal 1, 25, 50, 75, dan 100	Menganalisis performansi metode KNN terhadap pengaruh penetapan frekuensi dokumen minimal
2	Membandingkan nilai K pada KNN dengan rentang nilai 1 – 100.	Menganalisis performansi metode KNN terhadap pengaruh perubahan nilai K pada <i>dataset</i> ulasan film dalam Bahasa Inggris.
3	Membandingkan tingkat signifikansi pada <i>chi-square</i> dengan nilai 0.48, 0.49, 0.50, 0.51 dan 0.52.	Menganalisis performansi metode KNN terhadap perubahan tingkat signifikansi <i>chi-square</i> pada <i>dataset</i> ulasan film dalam Bahasa Inggris

4.1 Hasil Pengujian dan analisis skenario 1 (pengaruh penerapan frekuensi dokumen minimal)

Pengujian ini dilakukan untuk melihat pengaruh dari penetapan frekuensi dokumen minimal. Frekuensi dokumen minimal ini memiliki arti minimal nilai D_f yang ditentukan sesuai *threshold*. *Threshold* dokumen yang ditentukan pada penelitian kali ini adalah 1, 25, 50, 75, dan 100 dengan kombinasi ekstraksi fitur TF-IDF, metode KNN dengan nilai $K = 5$, dan seleksi fitur *chi-square* dengan tingkat signifikansi = 0.5. Pemilihan nilai $K = 5$ dan tingkat signifikansi 0.50 dipilih dikarenakan dalam penelitian yang dilakukan Indah Listiowarni [7], hasil paling optimum diperoleh jika nilai $K = 5$ dengan tingkat signifikansi 0.50. Hasil dari pengujian ini dapat dilihat pada Tabel 10.

Tabel 10. Perbandingan Performansi Skenario 1

Evaluasi performansi	Frekuensi Dokumen Minimal				
	1	25	50	75	100
Akurasi	51.00%	53.75%	66.50%	62.50%	62.75%
Precision	100.00%	82.61%	91.25%	82.89%	81.48%
Recall	02.00%	09.50%	36.50%	31.50%	33.00%
F1-measure	03.92%	17.04%	52.14%	45.65%	46.98%

Berdasarkan Tabel 10, hasil uji penerapan frekuensi dokumen minimal sebanyak 50 memiliki hasil evaluasi performansi terbaik jika dibandingkan dengan penerapan frekuensi dokumen minimal lainnya dengan nilai akurasi 66.50%, *precision* 91.25%, *recall* 36.50%, dan *F1-measure* 52.14%. Dari hasil pengujian ini, hasil frekuensi dokumen minimal sebanyak 50 memiliki hasil evaluasi performansi yang lebih baik dari frekuensi dokumen minimal 1 dan 25 dikarenakan pada *threshold* tersebut, hasil ekstraksi fitur atau jumlah kata yang akan dijadikan fitur terlalu banyak, sedangkan untuk frekuensi dokumen minimal 50 memiliki hasil evaluasi performansi yang lebih baik dari frekuensi dokumen minimal 75 dan 100 dikarenakan pada *threshold* tersebut, hasil ekstraksi fitur atau jumlah kata yang dijadikan fitur terlalu sedikit. Berikut adalah tabel perbandingan jumlah fitur yang dihasilkan frekuensi dokumen minimal dari masing masing *threshold* yang ditentukan.

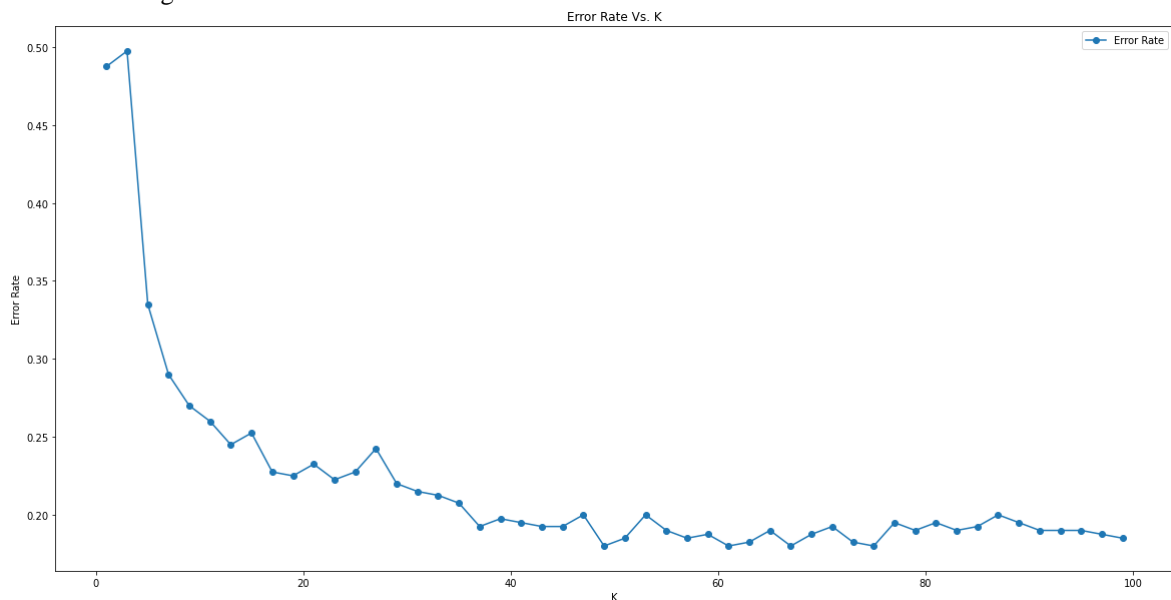
Tabel 11. Perbandingan Jumlah Fitur yang Dihasilkan

Frekuensi Dokumen Minimal	Jumlah Fitur yang Dihasilkan
1	15030
25	2776
50	1484
75	1003
100	735

Berdasarkan Tabel 11. Frekuensi dokumen minimal 1 dan 25 memiliki jumlah fitur yang lebih banyak jika dibandingkan dengan frekuensi dokumen minimal 50, sehingga terdapat fitur *noise* yang masih digunakan untuk melakukan prediksi dan mempengaruhi kemampuan sistem untuk memprediksi nilai positif. Hal ini dapat dibuktikan dengan nilai *recall* yang buruk pada *threshold* tersebut. Untuk frekuensi dokumen minimal 75 dan 100, jumlah fitur yang dihasilkan lebih sedikit jika dibandingkan dengan frekuensi dokumen minimal 50. Hal ini menyebabkan kemampuan sistem untuk memprediksi nilai positif menurun juga dikarenakan terdapat fitur yang mampu meningkatkan kemampuan sistem untuk memprediksi nilai positif terbuang. Hal ini dapat dibuktikan dengan nilai *recall* yang lebih kecil jika dibandingkan dengan frekuensi dokumen minimal 50.

4.2 Hasil pengujian dan analisis skenario 2 (pengaruh perubahan nilai K pada metode KNN)

Pengujian ini dilakukan untuk melihat pengaruh dari perubahan nilai K pada metode KNN dengan rentang nilai 1-100. Dikarenakan frekuensi dokumen minimal sebanyak 50 memberikan hasil evaluasi performansi yang paling baik pada skenario 1, skenario 2 akan menggunakan kondisi frekuensi dokumen minimal tersebut. Dengan masih menerapkan tingkat signifikansi = 0.50, hasil berupa grafik yang membandingkan antara *error rate* dengan nilai K adalah sebagai berikut.



Gambar 3. Grafik Perbandingan Error Rate dengan Nilai K

Berdasarkan Gambar 3, nilai K = 37 merupakan nilai K yang paling optimal. Hal ini dikarenakan *error rate* untuk nilai K yang lebih besar dari K = 37 tidak mengalami perubahan yang terlalu signifikan. Hal ini juga sesuai dengan *elbow method* untuk mencari nilai K yang paling optimal. Selain itu, penerapan nilai K = 37 juga menyebabkan hasil evaluasi performansi mengalami peningkatan yang cukup tinggi jika dibandingkan dengan nilai K = 5 yang digunakan pada skenario sebelumnya. Berikut adalah tabel perbandingan antara nilai K = 5 dengan nilai K = 37.

Tabel 12. Perbandingan Evaluasi Performansi Nilai K = 5 dengan Nilai K = 37

Evaluasi Performansi	Nilai K	
	5	37
Akurasi	66.50%	80.75%
Precision	91.25%	83.24%
Recall	36.50%	77.00%
F1-measure	52.14%	80.00%

Berdasarkan Tabel 12, nilai K pada metode KNN memiliki pengaruh terhadap hasil evaluasi performansi agar memperoleh hasil yang baik. Dari hasil pengujian ini, penerapan nilai K = 37 mampu meningkatkan kemampuan sistem untuk memprediksi nilai positif. Hal ini dapat dibuktikan dengan nilai *recall* yang jauh membaik jika dibandingkan dengan nilai K = 5. Dengan meningkatnya nilai *recall* juga mengakibatkan nilai *F1-measure* yang merupakan rata-rata harmonik antara *precision* dan *recall* ikut meningkat walaupun nilai *precision* mengalami sedikit penurunan. Selain itu, kemampuan sistem untuk memprediksi nilai benar positif dan nilai benar negatif juga menjadi meningkat. Hal ini dapat dibuktikan dengan meningkatnya nilai akurasi yang awalnya 66.50% menjadi 80.75%.

4.3 Hasil pengujian dan analisis skenario 3 (pengaruh perubahan tingkat signifikansi *chi-square*)

Pengujian ini dilakukan untuk melihat pengaruh dari perubahan tingkat signifikansi *chi-square* dengan nilai antara 0.48, 0.49, 0.50, 0.51, dan 0.52. Dikarenakan frekuensi dokumen minimal 50 memberikan hasil evaluasi performansi terbaik pada skenario 1, kondisi frekuensi dokumen minimal tersebut akan kembali digunakan pada skenario 3. Nilai $K = 37$ juga akan digunakan pada skenario 3 dikarenakan pada skenario 2 mampu menghasilkan evaluasi performansi yang cukup baik. Berikut adalah tabel perbandingan evaluasi performansi antar tingkat signifikansi.

Tabel 13. Perbandingan Evaluasi Performansi Tingkat Signifikansi

Evaluasi Performansi	Tingkat Signifikansi					Tanpa <i>Chi-square</i>
	0.48	0.49	0.50	0.51	0.52	
Akurasi	67.50%	77.25%	80.75%	79.25%	78.00%	75.75%
Precision	85.71%	87.07%	83.24%	77.99%	73.93%	71.72%
Recall	42.00%	64.00%	77.00%	81.50%	86.50%	85.00%
F1-measure	56.37%	73.78%	80.00%	79.71%	79.72%	77.80%

Berdasarkan Tabel 13, hasil uji penerapan seleksi fitur *chi-square* dengan tingkat signifikansi 0.50 merupakan model terbaik. Hal ini dikarenakan data latih yang digunakan seimbang, terdiri dari 800 data negatif dan 800 data positif sehingga evaluasi performansi yang dijadikan acuan adalah akurasi. Dari hasil pengujian ini, tingkat signifikansi 0.50 memiliki nilai akurasi yang lebih tinggi jika dibandingkan dengan tingkat signifikansi lainnya. Hal ini dikarenakan jika menerapkan tingkat signifikansi 0.48 dan 0.49, fitur yang dipilih menjadi lebih sedikit sehingga terdapat fitur yang mampu mempengaruhi hasil evaluasi performansi terbuang, sedangkan jika menerapkan tingkat signifikansi 0.51, 0.52, dan tanpa *chi-square*, fitur yang dipilih lebih banyak sehingga terdapat fitur *noise* yang masih digunakan untuk melakukan prediksi dan mengakibatkan hasil evaluasi performansi menjadi menurun. Berikut adalah tabel yang menyatakan perbandingan jumlah fitur diantara tingkat signifikansi 0.48, 0.49, 0.50, 0.51, dan 0.52.

Tabel 14. Perbandingan Jumlah Fitur

Jumlah Fitur	Tingkat Signifikansi					Tanpa <i>Chi-square</i>
	0.48	0.49	0.50	0.51	0.52	
Terpilih	819	1277	1484	1530	1563	1574
Terbuang	755	297	90	44	11	0

Berdasarkan Tabel 14, semakin kecil tingkat signifikansi yang diterapkan, maka semakin banyak fitur yang terbuang. Sebaliknya, semakin besar tingkat signifikansi yang diterapkan maka semakin sedikit juga fitur yang terbuang. Penggunaan fitur yang digunakan sangat mempengaruhi hasil evaluasi performansi. Semakin banyak fitur yang digunakan, maka kemampuan sistem dalam memprediksi nilai positif semakin tinggi. Hal ini dapat dibuktikan dengan peningkatan nilai *recall* pada Tabel 13. Akan tetapi, pada kondisi penerapan tingkat signifikansi 0.52 dan kondisi penerapan tanpa *chi-square* ternyata mengalami penurunan nilai *recall*. Hal ini menunjukkan diantara 11 fitur yang terbuang, terdapat fitur *noise* yang terbuang juga jika menerapkan tingkat signifikansi 0.52 pada Tabel 14. Dengan demikian, penggunaan seleksi fitur dengan tingkat signifikansi 0.52 perlu dilakukan jika menginginkan nilai *recall* tertinggi. Selain itu, berdasarkan Tabel 14 juga, semakin sedikit fitur yang digunakan, maka semakin tinggi rasio antara nilai benar positif dengan keseluruhan nilai yang diprediksi positif. Hal ini dapat dibuktikan dengan peningkatan *precision* pada Tabel 13. Akan tetapi, pada kondisi tingkat signifikansi 0.48, ternyata mengalami penurunan jika dibandingkan dengan tingkat signifikansi 0.49. Hal ini menunjukkan diantara 755 fitur yang terbuang, terdapat fitur yang mempengaruhi sistem untuk memprediksi nilai benar positif yang terbuang juga jika menerapkan tingkat signifikansi 0.48 pada Tabel 14. Dengan demikian, penggunaan seleksi fitur dengan tingkat signifikansi 0.49 perlu dilakukan jika menginginkan nilai *precision* tertinggi. Jika menginginkan keseimbangan antara *precision* dan *recall*, pilih tingkat signifikansi yang memiliki nilai *F1-measure* tertinggi. Berdasarkan Tabel 13, penggunaan seleksi fitur dengan tingkat signifikansi 0.50 dapat menjadi pilihan jika menginginkan keseimbangan antara *precision* dan *recall*.

5. Kesimpulan

Berdasarkan hasil pengujian yang dilakukan, dapat disimpulkan bahwa penggunaan frekuensi dokumen minimal sebanyak 50 merupakan terbaik diantara frekuensi dokumen minimal lainnya. Hal ini disebabkan frekuensi dokumen minimal dengan 1 dan 25 memiliki jumlah fitur yang lebih banyak jika dibandingkan dengan frekuensi dokumen minimal 50, sehingga terdapat fitur *noise* yang masih digunakan untuk melakukan prediksi dan mempengaruhi kemampuan sistem untuk memprediksi nilai positif dengan bukti nilai *recall* yang buruk. Untuk frekuensi dokumen minimal 75 dan 100, jumlah fitur yang dihasilkan lebih sedikit jika dibandingkan dengan frekuensi dokumen minimal 50. Hal ini menyebabkan kemampuan sistem untuk memprediksi nilai positif menurun juga dikarenakan terdapat fitur yang mampu meningkatkan kemampuan sistem untuk memprediksi nilai positif terbuang sehingga nilai *recall* yang dihasilkan lebih kecil dari frekuensi dokumen minimal 50. Selain frekuensi dokumen minimal, pencarian nilai K yang optimum pada KNN juga dapat mempengaruhi hasil evaluasi performansi. Berdasarkan hasil pengujian, nilai K yang optimum adalah 37. Hal ini dikarenakan jika menerapkan nilai K yang lebih besar dari 37, tidak terjadi perubahan *error rate* yang signifikan. Dengan diterapkan nilai K = 37 juga mampu meningkatkan hasil evaluasi performansi. Hal ini dapat dibuktikan dengan meningkatnya nilai akurasi, *recall*, dan *F1-measure*. Selain frekuensi dokumen minimal dan nilai K yang optimum, tingkat signifikansi pada seleksi fitur *chi-square* juga mempengaruhi hasil evaluasi performansi. Jika menerapkan tingkat signifikansi yang terlalu kecil, akan banyak fitur yang terbuang. Sebaliknya, jika menerapkan tingkat signifikansi yang terlalu besar, fitur yang terbuang sedikit. Oleh karena itu, pencarian tingkat signifikansi yang optimal perlu dilakukan. Berdasarkan hasil pengujian yang dilakukan, tingkat signifikansi 0.50 merupakan model yang memiliki hasil evaluasi performansi terbaik dengan nilai akurasi 80.75%, *precision* 83.24%, *recall* 77.00%, dan *F1-measure* 80.00%. Dengan menerapkan tingkat signifikansi 0.50 pada seleksi fitur *chi-square* juga mampu menghasilkan hasil evaluasi performansi yang lebih baik jika dibandingkan dengan tanpa menggunakan seleksi fitur *chi-square*. Hal ini dapat dibuktikan dengan peningkatan nilai akurasi sebesar 05.00%, *precision* sebesar 11.52%, dan *F1-measure* sebesar 02.20%. Meskipun terdapat penurunan kemampuan sistem untuk memprediksi nilai positif yang ditandai dengan menurunnya nilai *recall* sebesar 8%, rasio nilai benar positif dengan keseluruhan prediksi positif meningkat. Hal ini ditandai dengan meningkatnya nilai *precision* sebesar 11.52%. Dengan meningkatnya nilai *precision* juga mampu meningkatkan keseimbangan antara prediksi positif dengan negatif. Hal ini dapat dibuktikan dengan meningkatnya nilai *F1-measure* sebesar 02.20% dan akurasi sebesar 05.00%.

Saran untuk penelitian selanjutnya, penggunaan *k-fold cross validation* untuk memvalidasi keakuratan model yang dibangun serta bandingkan dengan metode klasifikasi lain.

REFERENSI

- [1] A. Kurniawan, Indriati, and S. Adinugroho, "Analisis Sentimen Opini Film Menggunakan Metode Naïve Bayes dan Lexicon Based Features," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 9, pp. 8335–8342, 2019.
- [2] Jatmiko Indriyanto, "ALGORITMA K-NEAREST NEIGHBOR UNTUK PREDIKSI NASABAH ASURANSI," p. 14, 2021, [Online]. Available: https://www.google.co.id/books/edition/ALGORITMA_K_NEAREST_NEIGHBOR_UNTUK_PREDI/EE0tEAAAQBAJ?hl=id&gbpv=0.
- [3] L. R. Dharmawan, I. Arwani, and D. E. Ratnawati, "Analisis Sentimen pada Sosial Media Twitter Terhadap Layanan Sistem Informasi Akademik Mahasiswa Universitas Brawijaya dengan Metode K-Nearest Neighbor," vol. 4, no. 3, pp. 959–965, 2020.
- [4] B. Stecanelia, "What is TF-IDF?," *monkeylearn*, 2019. What is TF-IDF? (accessed Nov. 14, 2020).
- [5] N. Octaviani Faomasi Daeli, "Sentiment Analysis on Movie Reviews Using Information Gain and K-Nearest Neighbor," *J. Data Sci. Its Appl.*, vol. 3, no. 1, pp. 1–007, 2020, doi:10.34818/JDSA.2020.3.22.
- [6] M. Maleki, N. Manshour, and T. Kayıkçioğlu, "A Novel Simple Method to Select Optimal k in k-Nearest Neighbor Classifier," vol. 15, no. 2, pp. 464–469, 2017.
- [7] I. Listiowarni and E. R. Setyaningsih, "Feature Selection Chi-Square dan K-NN pada Pengkategorian Soal Ujian Berdasarkan Cognitive Domain Taksonomi Bloom," *J. Komput. Terap.*, vol. 4, no. 1, pp. 21–30, 2018.
- [8] U. I. Larasati, M. A. Muslim, R. Arifudin, and A. Alamsyah, "Improve the Accuracy of Support Vector Machine Using Chi Square Statistic and Term Frequency Inverse Document Frequency on Movie Review Sentiment Analysis," *Sci. J. Informatics*, vol. 6, no. 1, pp. 138–149, 2019, doi:10.15294/sji.v6i1.14244.
- [9] K. Kumar, B. S. Harish, and H. K. Darshan, "Sentiment Analysis on IMDb Movie Reviews Using Hybrid Feature Extraction Method," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 5, no. 5, p. 109, 2019, doi:10.9781/ijimai.2018.12.005.
- [10] P. . ZAHARATUL FIKRINA, Teguh Bharata Adji, S.T., M.T., M. Eng., Ph.D ; Hanung Adi Nugroho, S.T., M. E., "KLAUSTERISASI TEKS BERBAHASA INDONESIA MENGGUNAKAN METODE TERM FREQUENCY AND INVERSE DOCUMENT FREQUENCY (TF-IDF) DAN KESAMAAN

- KOSINUS.”
- [11] S. A. Alasadi and W. S. Bhaya, “Review of data preprocessing techniques in data mining,” *J. Eng. Appl. Sci.*, vol. 12, no. 16, pp. 4102–4107, 2017, doi: 10.3923/jeasci.2017.4102.4107.
 - [12] F. E. Cahyanti, Adiwijaya, and S. Al Faraby, “On the Feature Extraction for Sentiment Analysis of Movie Reviews Based on SVM,” *2020 8th Int. Conf. Inf. Commun. Technol. ICoICT 2020*, 2020, doi: 10.1109/ICoICT49345.2020.9166397.
 - [13] J. Brownlee, “A Gentle Introduction to Statistical Hypothesis Testing,” 2018. <https://machinelearningmastery.com/statistical-hypothesis-tests/>.