MATICS : Jurnal Ilmu Komputer dan Teknologi Informasi ISSN : 1978-161X(p); 2477-2550(e)
Volume 9, No. 1(2017), pp 33-40 DOI : 10.18860/mat.v9i1.XXXX
Received : June 15th 2018; **Accepted** : July20th 2018 ; **Avalaible Online** : April 5th 2016

Document Similarity Measure Using Cosine Similarity Based on Class-Based Indexing

Syahroni Wahyu Iriananda, Muhammad Aziz Muslim, Harry Soekotjo Dachlan

Abstract— Report handling on "LAPOR!" system depends on the system administrator who manually reads every incoming report [3]. Read manually can lead to errors in handling complaints [4] if the data flow is very large and grows rapidly it can take at least three days and sensitive to inconsistencies [3]. In this study, the authors propose a model that can measure and identify the similarity of document reports computerized that can identify the similarity between the Query (Incoming) with Document (Archive). In this study, the authors employed term weighting scheme Class-Based Indexing, and Cosine Similarity to analyze document similarities. CoSimTFIDF, CoSimTFICF and CoSimTFIDFICF values are defined as feature sets for the text classification process using the K-Nearest Neighbor (K-NN) method. The optimum result evaluation with preprocessing employ Stemming and the best result of all features is 75% training data ratio and 25% test data on the CoSimTFIDF feature that is 84%. Value k = 5 has a high accuracy of 84.12%

Keywords— Complaints, Text Similarity, Class-Based Indexing, Cosine Similarity, K-Nearest Neighbor, LAPOR!

Abstrak— Pada Sistem "LAPOR!" penanganan laporan bergantung pada administrator sistem yang membaca secara manual setiap laporan yang masuk [3]. Hal ini dapat menyebabkan kesalahan dalam menangani keluhan [4], dan jika aliran datanya sangat besar dapat membutuhkan waktu minimal tiga hari, hal ini sensitif terhadap inkonsistensi [3]. Dalam penelitian ini penulis mengusulkan suatu model atau pendekatan yang dapat mengukur dan mengidentifikasi kemiripan dokumen laporan yang dilakukan secara terkomputerisasi yang dapat mengidentifikasi kemiripan antara Query dengan Document. Dalam penelitian ini penulis mempekerjakan skema pembobotan kata Class-Based Indexing, dan Cosine Similarity untuk menganalisa kemiripan dokumen. Nilai CoSimTFIDF, CoSimTFICF dan CoSimTFIDFICF kemudian ditetapkan sebagai set fitur untuk proses klasifikasi teks menggunakan metode K-Nearest Neighbor (K-NN). Hasil akurasi optimal dengan preprocessing Stemming dan hasil terbaik dari semua fitur adalah rasio data latih 75% dan data uji 25% pada fitur CoSimTFIDF yaitu 84%. Nilai k = 5 memiliki tingkat akurasi yang tinggi yaitu 84,12%

Kata Kunci— pengaduan, kemiripan teks, class-based indexing, cosine similarity, k-nearest neighbor.

Manuscript received March 15 June, 2018. (Write the date on which you submitted your paper for review.) This work was supported in part by Informatics Engineering Department of Maulana Malik Ibrahim Islamic State University. Paper titles should be written in uppercase and lowercase letters, not all uppercase. Avoid writing long formulas with subscripts in the title; short formulas that identify the elements are fine (e.g., "Nd–Fe–B"). Do not write "(Invited)" in the title. Full names of authors are preferred in the author field, but are not required. Put a space between authors' initials.

F. A. Author affiliated with the Electrical Engineering Departement of Brawijaya University, Malang, Indonesia (email roniwahyu@gmail.com)

S. B. Author affiliated with the Electrical Engineering Departement of Brawijaya University, Malang, Indonesia (e-mail: muh_aziz@ub.ac.id).

T. C. Author affiliated with the Electrical Engineering Departement of Brawijaya University, Malang, Indonesia (e-mail: harrysd@ub.ac.id)

I. PENDAHULUAN

Jumlah data laporan pengaduan dan opini publik yang masuk pada *platform* "LAPOR!" (Layanan Pengaduan dan Aspirasi Online Rakyat) dapat menjadi sumber informasi untuk mengukur kinerja pelayanan lembaga pemerintahan [1]. Rata-rata 900 laporan setiap hari, hanya 13% - 14% laporan diproses, sementara sekitar 86% tetap menjadi subyek yang belum diketahui dan diarsipkan. Saluran yang paling banyak digunakan adalah melalui SMS sekitar 80% - 90% laporan [2].

Administrator yang terbatas dan angka laporan pengaduan yang cukup tinggi menjadi penyebab utama kurangnya kualitas layanan terutama karakteristik daya tanggap (responsivitas) [2]. Penanganan laporan bergantung pada administrator sistem yang membaca secara manual setiap laporan yang masuk [3]. Hal ini dapat menyebabkan kesalahan dalam menangani keluhan [4], dan jika aliran datanya sangat besar dapat membutuhkan waktu minimal tiga hari, hal ini sensitif terhadap inkonsistensi [3]. Maka dari itu diperlukan solusi terhadap permasalahan analisis laporan pengaduan yang dapat membantu administrator "LAPOR!" dalam melakukan menentukan kategori sehingga analisis *big data* menjadi sangat penting [2].

Dalam penelitian ini penulis mengusulkan suatu model atau pendekatan yang dapat mengukur dan mengidentifikasi kemiripan dokumen laporan yang dilakukan secara terkomputerisasi yang dapat mengidentifikasi kemiripan antara *Query* dengan *Document*, berikutnya dilakukan klasifikasi terhadap *Query* untuk memprediksi kelas atau kategorinya berdasarkan nilai kemiripan terbesar atau yang mendekati nilai 1 (satu) antara *Query (Q)* dengan koleksi *Document (D)*. Penulis mempekerjakan skema pembobotan kata *Class-Based Indexing*, kemudian dikomparasi dengan skema pembobotan lainnya yaitu TFIDF dan TFICF. Nilai bobot TFIDF, TFICF dan TFIDFICF kemudian dikonversi ke dalam koordinat

kartesius dan dihitung kemiripannya menggunakan fungsi *Cosine Similarity* untuk menganalisa kemiripan dokumen teks dengan cara mendapatkan kemiripannya dengan cara mengukurnya dalam bentuk jarak vektor kemiripan. Nilai *Cosine Similarity* dari ketiga skema pembobotan tersebut (CoSimTFIDF, CoSimTFICF, CoSimTFIDFICF) kemudian ditetapkan sebagai set fitur untuk proses klasifikasi. Berikutnya dilakukan proses klasifikasi teks menggunakan metode *K-Nearest Neighbor (K-NN)* untuk klasifikasi dokumen dan memprediksi kategori dokumen baru berdasarkan fitur-fitur tersebut.

Penelitian ini bertujuan untuk mengidentifikasi dan mengevaluasi kemiripan teks menggunakan metode TF-IDF-ICF *(Class Indexing Based)* dan *Cosine Similarity.*

II. KAJIAN PUSTAKA

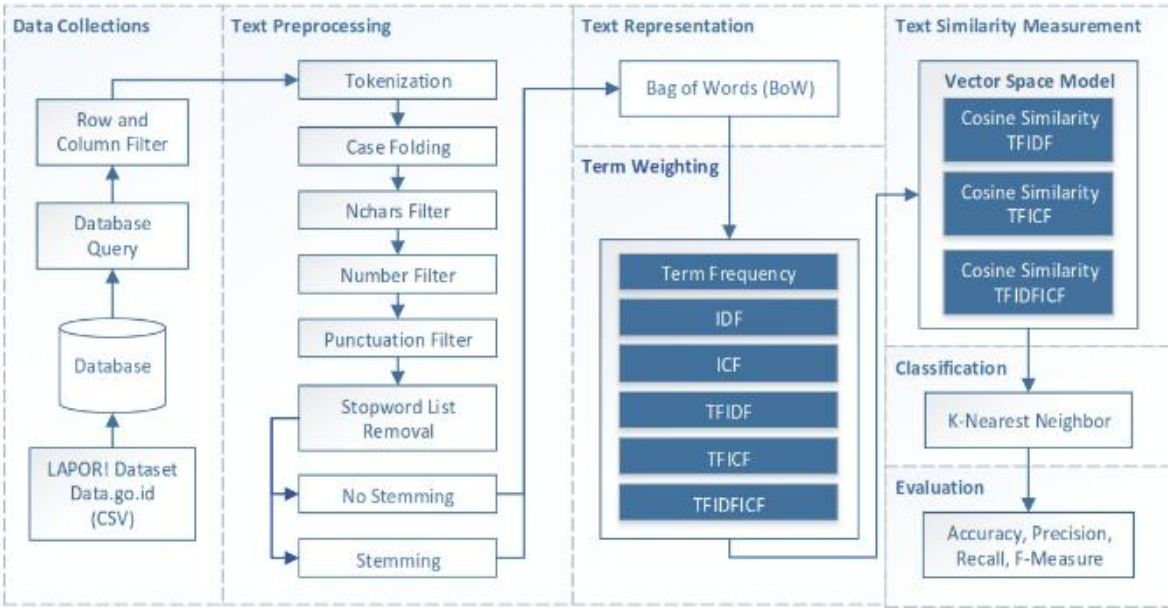
Suatu kata disebut mirip secara Leksikal, yaitu ketika suatu kata memiliki urutan karakter yang sama. Dan ketika suatu kata memiliki makna yang sama maka disebut mirip secara Semantik [5]. Penelitian [6] dengan memanfaat kan TF.IDF.ICF untuk klasifikasi dokumen pengaduan (e-complaint) mahasiswa menggunakan *Centroid Based Classifier*, dikombinasikan dengan TF.IDF.ICF, *Cosine Similarity* dan *Class Feature Centroid*. [7] Melakukan kategorisasi ide kreatif pada suatu perusahaan menggunakan algoritma K-NN dan TF.IDF.ICF. [8] Mengklasifikasikan pengaduan SambatOnline Kota Malang menggunakan algoritma K-NN, *Cosine Similarity* dan *Chi Square* daripada TFIDF. [9] Menggunakan algoritma K-NN dan seleksi fitur TFIDF, dan *Categorical Proporsional Difference (CPD)*. Dataset yang sama digunakan [10] dengan mempekerjakan algoritma NW-K-NN, term weighting TFIDF filter N-Gram, dan Unigram pada preprocessing. Hasil eksperimen [11] menunjukkan bahwa klasifikasi teks dapat digunakan untuk mengevaluasi kualitas layanan dengan data teks dari penanganan keluhan pelanggan (complaint). Metode ini dapat memecahkan masalah evaluasi otomatis dalam manajemen penanganan keluhan pelanggan. [11]

III. METODE PENELITIAN

Secara garis besar kerangka solusi masalah dapat dilihat pada Gambar 1, yang terdiri dari 7 (tujuh) proses utama, yaitu Pengumpulan data *(Data Collecting)*, pra-proses teks *(Text Preprocessing)*, representasi teks *(Text Representation)*, seleksi fitur *(Feature Selection)* mencakup *Term Weighting* pada umumnya (TFIDF) dan *Class Based Indexing* (ICF), klasifikasi, dan evaluasi.

Penelitian ini menggunakan tiga skema pembobotan untuk dikomparasi dan dievaluasi untuk mendapatkan pembobotan yang yang memiliki hasil paling optimal. Pengujian yang dilakukan adalah eksperimen pada proses "Preprocessing" yaitu melalui sub-proses *Stemming* dan tidak menggunakan Stemming. Eksperimen dengan (term weighting) yang berbeda yaitu menggunakan TF-IDF, TF-ICF dan TF-IDF-ICF berikut dengan ekperimen variasi *Cosine Similarity* berdasarkan masing-masing pembobotan kata tersebut. ekperimen dengan variasi jumlah data, dan variasi jumlah data training dan data testing. Kemudian pada hasil akhir akan dievaluasi pengaruh kinerja keduanya.

Metode yang digunakan untuk menganalisa kemiripan antara laporan pengaduan yang baru masuk *(Query)* dengan laporan yang telah diproses administrator *(Document)* adalah *Cosine Similarity*. Hasil *term-weighting* dengan TF-IDF, TF-ICF dan TF-IDF-ICF kemudian dikonversi ke dalam koordinat kartesius dan dihitung menggunakan fungsi *Cosine Similarity* untuk mendapatkan sudut kemiripannya *dan* mengukur jarak vector. Berikutnya dilakukan proses klasifikasi teks berdasarkan fitur *Cosine Similarity* TF-IDF (CoSimTFIDF), TF-ICF (CoSimTFICF), TF-IDF-ICF (CoSimTFIDFICF) yang menggunakan skema pembobotan yang berbeda-beda. Semakin besar nilai ketiga fitur cosine similarity yaitu mendekati nilai 1 (satu), maka semakin mirip suatu *Query (q)* dengan



Gambar 1 Kerangka Konsep Kemiripan Dokumen Dengan Cosine Similarity Berbasis Class-Based Indexing

koleksi *Document (d).* Metode K-Nearest Neighbor (KNN) dipilih untuk melakukan klasifikasi untuk memprediksi kategori Query.

A. Class Based Indexing (ICF)

Skema pembobotan berbasis kategori diusulkan [12]. Penelitian ini memperkenalkan Frekuensi Kategori Terbalik *(Inverse Category Frequency)* dalam skema pembobotan istilah untuk tugas klasifikasi teks. Dua konsep didefinisikan sebagai: *Frekuensi Kategori (CF)* yaitu jumlah kategori di mana istilah *(t)* muncul dan *Frekuensi Kategori Terbalik (ICF)* yang formulanya mirip dengan IDF. [12]. Konsep *Class Based Indexing (ICF)* berikutnya dikembangkan oleh [13], Pada penelitian tersebut dikenalkan Metode pengindeksan otomatis menggunakan kombinasi berbasis dokumen (IDF) dan kelas/kategori (ICF) yang lebih baik yaitu TFIDFICSdF (Term Frequency Inverse Document Frequency Inverse Class Space Density Frequency), dimana (d) merupakan *density* atau kepadatan anggota term atau kemunculan suatu term dalam kategori atau kelas tertentu. Pada skema pembobotan istilah IDF hanya memperhatikan kemunculan term pada kumpulan dokumen dan melakukan pembobotan berbasis dokumen tanpa memperhatikan kelas/kategori yang merupakan induk dokumen tersebut. Sementara pendekatan menggunakan *Inverse Class Frequency (ICF)* memperhatikan kemunculan term pada kumpulan kategori/kelas. Term yang jarang muncul pada banyak kelas adalah term yang bernilai untuk klasifikasi. Semakin jarang kemunculan term tersebut, maka nilainya akan semakin besar atau mendekati nilai 1 (satu), dan sebaliknya semakin sering kemunculan term tersebut maka nilainya semakin kecil atau mendekati nilai 0 (nol). Kepentingan tiap term diasumsikan memilki proporsi yang berkebalikan dengan jumlah kelas yang mengandung term. Penentuan indeks yang akurat juga bergantung pada kepentingan term terhadap kelas atau kelangkaan term pada keseluruhan kelas *(rare term)*. Sehingga dibutuhkan term weighting berbasis kelas yang dinamakan inverse class frequency (ICF). Namun ICF hanya memperhatikan term yang ada pada kelas tanpa memperhatikan jumlah term dalam dokumen yang menjadi anggota kelas. Formula ICF dihitung dengan formula:

$$ICFLog_i = Log_2\left(\frac{|C|}{cf_i}\right) \quad (1)$$

Dimana C adalah jumlah seluruh kelas/kategori dalam koleksi cf_i adalah jumlah kelas/kategori yang mengandung term t_i

B. Persiapan dan Pengolahan Data

Proses pengumpulan data diawali dengan mengunduh data laporan pengaduan aplikasi "LAPOR!" yang berupa data teks. Data ini didapatkan dengan cara mengunduh data yang telah dipublikasikan pada portal berbagi data publik yaitu http://data.go.id. Kemudian data yang masih dalam bentuk file CSV tersebut di ekspor ke dalam database MySQL, tujuannya adalah untuk mempermudah peneliti melakukan eksperimen.

Penggalian data teks sangat bergantung pada data yang digunakan. Pada penelitian ini digunakan kumpulan data (dataset) laporan pengaduan masyarakat. Dataset utama merupakan data platform "LAPOR!" yang telah dipublikasikan. Berikut ini merupakan dataset yang digunakan dalam penelitian ini:

1) Dataset LAPOR!

Dataset utama merupakan data sekunder yang merupakan data aliran laporan pengaduan pada platform "LAPOR!" sejak tahun 2012 hingga Januari 2015. Data ini secara bebas dapat didownload pada situs berbagi data pemerintahan secara terbuka *(Open Government Indonesia)* yaitu data.go.id. Berikut ini merupakan contoh data pengaduan "LAPOR!" seperti pada Tabel 1

Tabel 1 Tabel data sampel laporan pengaduan "LAPOR!"

JudulLaporan	IsiLaporan	Kategori
Peserta KKS Belum Mendapat KIP untuk Anak Sekolahnya (Kuningan, Jabar)	38fcob45574000 saya penerima kks.tetapi saya tdk mendapat kartu indonesia pintar padahal anak saya sudah sekolah semua.bagaimana?	Kartu Indonesia Pintar (KIP)

2) Query Database (Input Data)

Query Database merupakan tahap awal daripada penelitian ini. Istilah *query database* yang dimaksud adalah suatu query bahasa database SQL yang digunakan pada server MySQL untuk memilih sekumpulan data. Data ini digunakan sebagai data masukan (input data) pada proses awal penelitian. Peneliti perlu melakukan pembatasan data, dalam rangka efisiensi waktu eksperimen, mengurangi dimensi data yang *irrelevant* dan data *noise*, serta untuk proses pengujian dan evaluasi penelitian yang efektif, maka dalam penelitian ini menggunakan data laporan yang sesuai dengan kriteria berikut ini: a) Data laporan masuk pada tahun 2015, b) Melalui kanal SMS atau aplikasi mobile c) Status laporan "Selesai" d) Memperhatikan prioritas urusan dalam pemerintahan yaitu: 1) Pendidikan 2) Kesehatan 3) Inftrastruktur . Query data dengan kriteria tersebut, menghasilkan jumlah data sekitar 7.134 baris data dan 82 kategori. Kriteria pada poin empat dipilih dengan memperhatikan jumlah data terbanyak pada kategori yang terkait dengan prioritas urusan yang terdapat pada poin empat, Dengan demikian jumlah baris data secara drastis berkurang menjadi 2.825 baris data dan jumlah kategori berkurang dari 82 kategori menjadi 8 kategori seperti yang disajikan pada Tabel 2

Tabel 2 Query Database Berdasarkan Kategori

Kategori	*Jumlah Data*
Kartu Indonesia Pintar (KIP)	1.389
Infrastruktur	574
Kartu Indonesia Sehat (KIS)	493
Pendidikan	131
Kesehatan	111
BPJS Kesehatan	79
Pendidikan Dasar dan Menengah (Dikdasmen)	35
Pelayanan Kesehatan	13
Total Dokumen	**2.825**

Penelitian ini menggunakan beberapa scenario ekperimen yang salah satunya adalah variasi dataset yang terdapat pada Tabel 3. Skenario ini bertujuan

untuk untuk menginvestigasi pengaruh jumlah baris data terhadap proses terkait.

Tabel 3 Tabel Partisi Data Document (D) dan Query (Q)

Seri Dataset	90% (D)	10% (Q)	Jumlah Data
Dataset25	22	3	25
Dataset50	45	5	50
Dataset75	67	8	75
Dataset100	90	10	100
Dataset200	180	20	200
Dataset300	270	30	300
Dataset400	360	40	400

Pada variasi dataset tabel 3 tersebut setiap anggota data dalam dataset tersebut dipilih secara acak *(random sampling)* sebanyak Dataset(n) dari dataset "LAPOR!" pada tabel 2. Pemilihan dataset secara acak dilakukan pada proses awal sebelum proses preprocessing dilakukan. Dataset tersebut memiliki anggota atau jumlah baris data sesuai dengan yang tertera pada nama dataset.

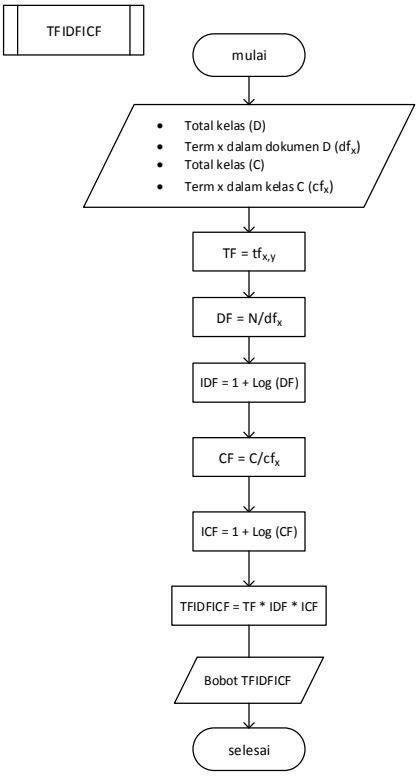
3) Data Partition

Sebagai bentuk simulasi dan eksperimen dalam penelitian ini dilakukan *Data Partition* atau partisi data. Ini dilakukan dengan cara membagi jumlah keseluruhan baris data dalam dataset pada tabel 3 menjadi dua bagian yaitu **1) Dataset *Documents (D)*** sebesar 90%, **2) Dataset *Query (Q)*** sebesar 10% seperti pada tabel 3 Setelah dilakukan proses Preprocessing dengan stemming maupun tanpa stemming selanjutnya dataset ini dibagi menjadi dua bagian yaitu 90% untuk dataset dokumen (D) dan 10% digunakan untuk dataset query (Q). Pembagian data juga dilakukan secara acak *(random sampling)* dengan demikian didapatkan anggota data seperti pada tabel 3

Pada dataset dokumen (D) kolom kategori masih tetap digunakan pada dataset, namun pada dataset query (Q) kolom kategori tersebut tidak dicantumkan. Keduanya digunakan untuk simulasi dan ekperimen, dimana hanya satu anggota dataset (Q) dibandingkan dengan banyak anggota dalam dataset (D).

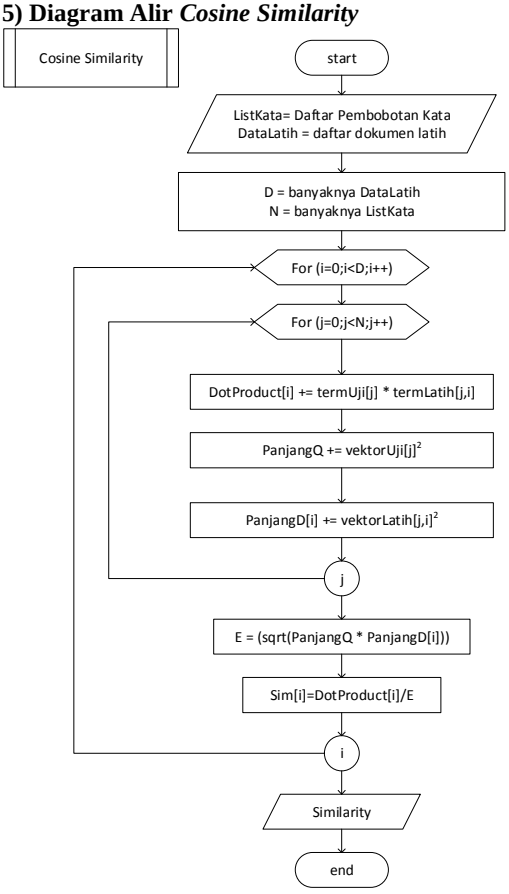
4) Diagram Alir Term Weighting

Pada Gambar 2, proses TFIDFICF merupakan gabungan dari proses pembobotan kata terhadap dokumen (TFIDF) dan pembobotan kata terhadap kategori (TFICF) dimana proses ini diawali dengan cara mendapatkan jumlah total dokumen (D), jumlah kata (x) yang terkandung pada dokumen (df_x) dan jumlah total kategori (C), jumlah kata (x) yang terkandung pada kelas atau kategori (cf_x). Setelah didapatkan jumlah total D dan C, selanjutnya adalah mendapatkan Term Frequency (TF), dan Document Frequency (DF), dan Class Frequency (CF). Setelah diketahui nilai TF, DF dan CF berikutnya adalah mendapatkan nilai inverse daripada DF (IDF) dan inverse daripada CF (ICF). Dari kombinasi bobot frekuensi tersebut menghasilkan nilai bobot TFIDF, TFICF, dan TFIDFICF yang akan digunakan untuk proses selanjutnya yaitu normalisasi menggunakan Cosine Similarity.



Gambar 2. Diagram Alir Sub Sistem *Term Weighting*

5) Diagram Alir *Cosine Similarity*



6)

Gambar 3. Diagram Alir Sub Sistem *Cosine Similarity*

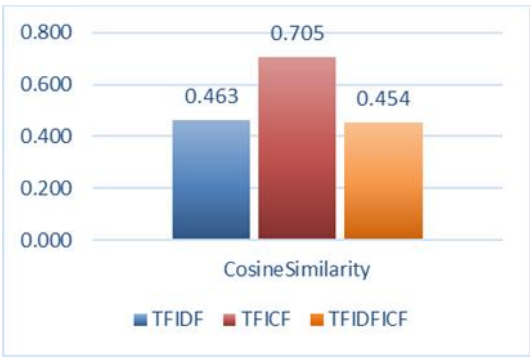
IV. HASIL DAN PEMBAHASAN

1) Hasil Pengukuran Cosine Similarity

Didiskusikan bagaimana pengukuran cosine similarity antara Query (Q) dengan Documents (D). Sebagaimana telah diketahui bahwa Cosine Similarity ini merupakan perkalian product (Dot Product) dari kedua vektor Q dan vektor D. Rumus cosine similarity menggunakan persamaan (2.7). Pada penelitian ini terdapat beberapa tahapan dalam pengukuran Cosine Similarity yang antara lain: **a)** Menghitung bobot kata (term weighting) pada *Documents* (D) untuk setiap skema pembobotan yaitu TFIDF, TFICF, dan TFIDFICF. **b)** Membentuk vektor Q dan D untuk setiap skema pembobotan kata. **c)** Menentukan data uji atau *query* (Q) yang ingin dilakukan pengujian dengan semua *documents* (D). Contoh data uji (Q). **d)** Menghitung bobot kata (term weighting) Query (Q) **e)** Perkalian dot vector Q dan vector D *(inner product)* **f)** Menghitung panjang vector Q *(magnitude)* **g)** Menghitung panjang vector D (magnitude) **h)** Menghitung *Cross Product* |Q| dan |D|

Perbandingan Hasil Cosine Similarity Berdasarkan Term Weighting

Setelah dilakukan serangkaian perhituangan manual cosine similarity terhadap ketiga pembobotan TFIDF, TFICF, dan TFIDFICF dapat diketahui hasil cosine similarity diantaranya dibandingkan dalam gambar 4. Dan dalam perhitungan cosine simiality didapatkan bahwa hasil rekomendasi dokumen berdasarkan nilai Cosine Similarity dengan skema pembobotan TFIDF, TFICF, dan TFIDFICF adalah dokumen D5 dengan hasil nilai cosine terbesar adalah 0,705 atau 70,5% berdasarkan pembobotan TFICF.



Gambar 4 Grafik Perbandingan Cosine Similarity Berdasarkan Skema Pembobotan

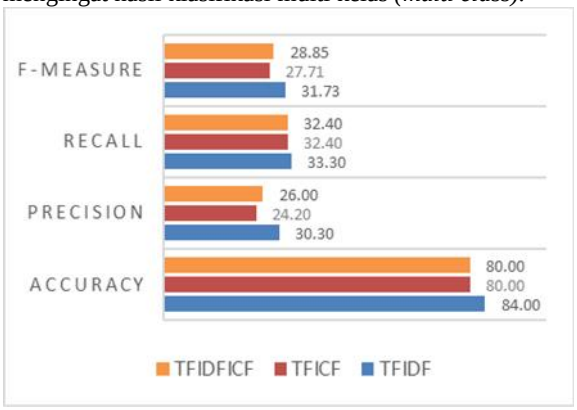
2) Hasil Eksperimen Variasi Preprocessing

Tabel 5 Tabel Skenario Pengujian Variasi Preprocessing

Skenario	*Jumlah Data*		*Hasil Evaluasi (%)*			
Term Weighting	*Latih*	*Uji*	*A*	*P*	*R*	*F1*
TFIDF	***75***	***25***	**84.00**	**30.30**	**33.30**	**31.73**
TFICF	***75***	***25***	80.00	24.20	32.40	27.71
TFIDFICF	***75***	***25***	80.00	26.00	32.40	28.85
TFIDF	***90***	***31***	46.15	17.71	19.82	18.71
TFICF	***90***	***31***	**58.06**	**31.88**	**21.13**	**25.42**
TFICFICF	***90***	***31***	45.16	18.14	17.21	17.66

Pada ekperimen ini bertujuan untuk mengevaluasi kinerja term weighting Class Indexing Based (TFIDFCF) dibandingkan dengan kinerja term weighting TFIDF, dan TFICF. Dataset yang digunakan adalah Dataset200, Dengan rasio perbandingan data training 75% dan data testing 25%. Menggunakan enam kategori kemudian diberikan label *(class)* sebagai nama lain *(alias)* ditunjukkan pada Tabel berikut:

Berikut adalah hasil ekperimen berdasarkan pengujian dengan variasi preprocessing stemming dan tanpa stemming. Evaluasi yang digunakan adalah Accuracy (A), Precision (P), Recall (R) dan F1-Measure (F1) menggunakan model macro average, ini digunakan mengingat hasil klasifikasi multi kelas *(multi class)*.



Gambar 5 Grafik evaluasi pengujian dengan proses stemming (dalam persen)

3) Pengujian Variasi Bobot Fitur

Tabel 6 Hasil Pengujian Bobot Fitur TFIDF

Evaluasi	Jumlah Dataset			
	100	**200**	**300**	**400**
A	69.23	**84.00**	60.53	63.33
P	18.89	**30.30**	17.36	21.22
R	18.89	**33.33**	18.89	21.15
F	18.89	**31,70**	18.00	18.47

Berdasarkan hasil yang tertera pada tabel 6, nilai akurasi terbaik untuk CoSimTFIDF adalah pada Dataset200 yaitu 84% dengan F-Measure 31,70%, kemudian Dataset100 dengan akurasi 69,23% namun nilai F-Measure cukup rendah yaitu 18,89%, sedangkan nilai F-Measure tertinggi didapatkan dari Sehingga didapatkan Dataset200 memiliki hasil performansi klasifikasi K-NN terbaik untuk klasifikasi laporan pengaduan dengan fitur CoSimTFIDF. Tingkat presisi dan nilai recall pada Dataset200 juga menunjukkan hasil dengan nilai tertinggi diantara dataset yang lainnya.

Tabel 7 Hasil Pengujian Bobot Fitur TFICF

Evaluasi	Jumlah Dataset			
	100	**200**	**300**	**400**
A	76.92	**80.00**	71.05	61.67
P	12.82	24.17	**28.24**	13.57
R	16.67	**32.41**	20.37	19.96
F	14.49	**27.68**	19.72	16.00

Berdasarkan hasil yang tertera pada tabel 7, nilai akurasi terbaik untuk CoSimTFICF adalah pada Dataset200 yaitu 80% dengan F-Measure 27,68%, kemudian Dataset100 dengan akurasi 76,92% namun nilai F-Measure cukup rendah yaitu 14,49%, Sehingga

didapatkan Dataset200 memiliki hasil performansi klasifikasi K-NN terbaik untuk klasifikasi laporan pengaduan dengan fitur CoSimTFICF. Tingkat presisi juga menunjukkan hasil yang cukup baik dengan nilai 24,17% dan nilai recall pada Dataset200 juga menunjukkan hasil dengan nilai tertinggi diantara dataset yang lainnya yaitu 32,41%.

Tabel 8 Hasil Pengujian Bobot Fitur TFIDFICF

Evaluasi	Jumlah Dataset			
	100	**200**	**300**	**400**
A	69.23	**80.00**	60.53	63.33
P	12.50	**25.99**	22.80	19.52
R	15.00	**32.41**	19.63	21.15
F	13.64	**28.7**	18.86	18.86

Berdasarkan hasil yang tertera pada tabel 8, nilai akurasi terbaik untuk CoSimTFIDFICF adalah pada Dataset200 yaitu 80% dengan F-Measure 28,7%, kemudian Dataset100 dengan akurasi 69,23% namun nilai F-Measure cukup rendah yaitu 13,64%, Sehingga didapatkan Dataset200 memiliki hasil performansi klasifikasi K-NN terbaik untuk klasifikasi laporan pengaduan dengan fitur CoSimTFICF. Tingkat presisi juga menunjukkan hasil yang cukup baik dengan nilai 25,99% dan nilai recall pada Dataset200 juga menunjukkan hasil dengan nilai tertinggi diantara dataset yang lainnya yaitu 32,41%.

4) Akurasi Proses Klasifikasi Berdasarkan Nilai k

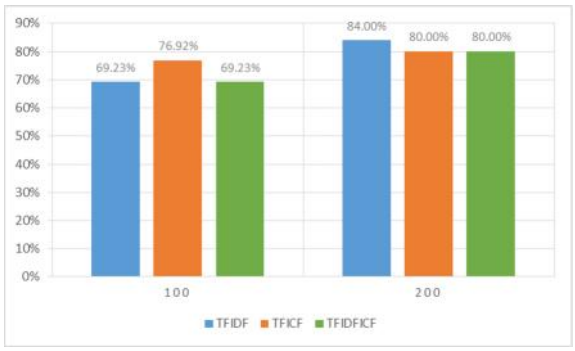
Ekperimen menggunakan Dataset200 dengan proses preprocessing menggunakan Stemming dan fitur CoSimTFIDF.

Tabel 9 Akurasi (%) KNN Berdasarkan Nilai k

Nilai k / *Rasio*	*1*	*2*	*3*	*4*	*5*
25/75	75.71	80.00	78.57	80.00	**80.00**
75/25	**83.33**	**83.33**	**83.33**	**83.33**	**83.33**
40/60	67.86	80.36	82.14	80.36	**82.14**
60/40	76.32	**84.21**	**84.21**	**84.21**	**84.21**

Dari pengujian yang dilakukan, tercatat hasil akurasi algoritma K-NN dengan pengujian rasio data latih 60% dan data uji 40% memiliki tingkat akurasi cukup tinggi terlihat dari hasil nilai k=1 cukup rendah, namun meningkat tajam 8% saat ujicoba nilai k=2 sampai k-5 dengan nilai stabil dan sama yaitu 84,21%, sedangkan pada pengujian ini ditemukan bahwa rasio data latih 75% dan data uji 25% menghasilkan nilai akurasi 83,33% lebih rendah 6,7% dari percobaan dengan variasi preprocessing pada tabel 5.39 yaitu 84%. Hal ini sangat mungkin terjadi karena pengambilan sampel data latih dan data uji digunakan adalah metode *random sampling.* Dalam pengujian ini dapat dilihat bahwa dengan nilai k=5 seluruh variasi rasio data latih dan data uji memiliki nilai maksimal daripada nilai k lainnya. Dengan demikian dapat disimpulkan berdasarkan pengujian yang telah dilakukan bahwa nilai k=5 merupakan nilai yang optimal

5) Hasil Perbandingan Akurasi KNN Berdasarkan Fitur Dan Dataset



Gambar 6 Akurasi KNN Berdasarkan Fitur Dan Dataset

Ekperimen dilakukan dengan menentukan dataset yang digunakan yaitu Dataset100 dan Dataset200. Dengan rasio perbanding antara data training 75% dan data testing 25% (75/25). Nilai k yang digunakan adalah k=5. Pada pengujian pertama fitur yang digunakan hanya fitur Cosine Similarity yang berbasis term weighting TFIDF, kemudian pada pengujian berikutnya digunakan fitur Cosine Similarity berbasis TFICF, berikutnya CoSimTFIDFICF. Hasil akurasi terbaik yang telah dicapai adalah menggunakan fitur Cosine Similarity berbasis TFIDF (CoSimTFIDF) yaitu 84% pada Dataset200 meningkat 4% dari kedua fitur Cosine Similarity TFICF dan TFIDFICF. Sedangkan pada Dataset100 diperoleh nilai akurasi terbaik dengan menggunakan fitur CoSimTFICF yaitu 76,92% meningkat sekitar 6% dari kedua fitur lainnya

6) Hasil Akurasi Dengan Variasi Rasio Data

Pada eksperimen ini menggunakan Dataset200 dengan variasi proses preprocessing menggunakan stemming dan tanpa stemming. Sebagaimana yang telah ditemukan pada tabel 5.43 dimana nilai K yang memiliki hasil optimal adalah k=5, maka ditetapkan pada pengujian ini klasifikasi K-NN menggunakan nilai k=5. Rasio perbandingan data latih dan data uji yang bervariasi guna mendapatkan hasil yang bervariasi. Berikut ini merupakan hasil pengujian akurasi berdasarkan rasio data dan fitur pada tabel 5.45

Tabel 10 Akurasi Pada variasi Term Weighting

Pre processing	Rasio (%)	Akurasi (%)		
		CoSim TFIDF	CoSim TFICF	CoSim TFIDFICF
Dengan Stemming	25:27	66.67	66.67	66.67
	75:25	**84.00**	**80.00**	**80.00**
	40:60	66.67	78.33	71.67
	60:40	60.00	75.00	70.00
Tanpa Stemming	25:27	54.50	64.86	60.36
	75:25	54.05	55.41	55.41
	40:60	52.81	60.67	60.67
	60:40	55.46	57.98	57.98

Berdasarkan hasil tersebut ditemukan bahwa hasil akurasi optimal dengan preprocessing Stemming dan

hasil terbaik dari semua fitur adalah rasio data latih 75% dan data uji 25% pada fitur Cosine Similarity berbasis term weighting TFIDF yaitu 84%. Kemudian fitur CoSimTFICF dengan rasio data latih 40% dan data uji 60%

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Dalam hasil pengujian yang telah dilaksanakan, ditemukan bahwa **1)** Skema *term weighting* TFIDF memiliki pengaruh yang signifikan terhadap akurasi klasifikasi. **2)** Pengujian dengan variasi proses stemming menggunakan fitur *Cosine Similarity* berbasis TFIDF (CoSimTFIDF) dengan mempekerjakan 75 data latih dan 25 data uji ini menghasilkan performa algoritma K-NN terbaik yaitu akurasi 84%, dengan tingkat presisi 30,3%, recall 33,3%, dan f-measure 31,73%. Hasil ini lebih baik 35% daripada preprocessing tanpa stemming yaitu sekitar 58%. **3)** Pengujian untuk menginvestigasi nilai k=1,2,3,4 dan 5 dengan 100 data latih dan uji dengan variasi rasio data latih dan data uji yang berbeda-beda adalah nilai k=5. Nilai akurasi terbaik diperoleh adalah rasio 60:40 yaitu 84,21%.

B. Saran

Beberapa hal yang dapat dikembangkan untuk penelitian selanjutnya dalam lingkup yang sama antara lain: **1)** Sebaiknya dilakukan penambahan variasi preprocessing yaitu stopword list berbeda bahasa misalkan Bahasa Sunda, Basaha Jawa, Bahasa Slang/gaul dan sebagainya. **2)** Sebaiknya juga dapat digunakan teknik *Cross Validation* untuk mendapatkan rasio data latih dan data uji K-NN yang proporsional.

VI. DAFTAR PUSTAKA

[1] A. Sofyan And S. Santosa, "Text Mining Untuk Klasifikasi Pengaduan Pada Sistem Lapor Menggunakan Metode C4.5 Berbasis Forward Selection," *Cyberku J.*, Vol. 12, No. 1, Pp. 8–8, 2016.

[2] I. Surjandari, "Application of Text Mining for Classification of Textual Reports: A Study of Indonesia's National Complaint Handling System," in *6th International Conference on Industrial Engineering and Operations Management (IEOM 2016)*, Kuala Lumpur, Malaysia.

[3] A. Fauzan and M. L. Khodra, "Automatic multilabel categorization using learning to rank framework for complaint text on Bandung government," in *2014 International Conference of Advanced Informatics: Concept, Theory and Application (ICAICTA)*, 2014, pp. 28–33.

[4] S. Tjandra, A. A. P. Warsito, and J. P. Sugiono, "Determining citizen complaints to the appropriate government departments using KNN algorithm," in *2015 13th International Conference on ICT and Knowledge Engineering (ICT Knowledge Engineering 2015)*, 2015, pp. 1–4.

[5] W. H. Gomaa and A. A. Fahmy, "A Survey of Text Similarity Approaches," *Int. J. Comput. Appl.*, vol. 68, no. 13, pp. 13–18, 2013.

[6] M. A. Rosid, G. Gunawan, and E. Pramana, "Centroid Based Classifier With TF – IDF – ICF for Classfication of Student's Complaint at Appliation E-Complaint in Muhammadiyah University of Sidoarjo," *J. Electr. Electron. Eng.-UMSIDA*, vol. 1, no. 1, pp. 17–24, Feb. 2016.

[7] R. R. M. Putri, R. Y. Herlambang, and R. C. Wihandika, "Implementasi Metode K-Nearest Neighbour Dengan Pembobotan TF.IDF.ICF Untuk Kategorisasi Ide Kreatif Pada Perusahaan," *J. Teknol. Inf. Dan Ilmu Komput.*, vol. 4, no. 2, pp. 97–103, May 2017.

[8] C. F. Suharno, M. A. Fauzi, and R. S. Perdana, "Klasifikasi Teks Bahasa Indonesia Pada Dokumen Pengaduan Sambat Online Menggunakan Metode K-Nearest Neighbors (K-NN) dan Chi-Square," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput. Vol 1 No 10 2017*, Jul. 2017.

[9] N. H. A. Sari, M. A. Fauzi, and P. P. Adikara, "Klasifikasi Dokumen Sambat Online Menggunakan Metode K-Nearest Neighbor dan Features Selection Berbasis Categorical Proportional Difference," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput. Vol 2 No 8 2018*, Oct. 2017.

[10] A. A. Prasanti, M. A. Fauzi, and M. T. Furqon, "Klasifikasi Teks Pengaduan Pada Sambat Online Menggunakan Metode N-Gram dan Neighbor Weighted K-Nearest Neighbor (NW-KNN)," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput. Vol 2 No 2 2018*, Aug. 2017.

[11] S. Dong and Z. Wang, "Evaluating service quality in insurance customer complaint handling throught text categorization," in *2015 International Conference on Logistics, Informatics and Service Sciences (LISS)*, 2015, pp. 1–5.

[12] D. Wang and H. Zhang, "Inverse-Category-Frequency based supervised term weighting scheme for text categorization," *J. Inf. Sci. Eng.*, vol. 29, no. 2, pp. 209–225, Dec. 2010.

[13] F. Ren and M. G. Sohrab, "Class-indexing-based term weighting for automatic text classification," *Inf. Sci.*, vol. 236, pp. 109–125, Jul. 2013.

First A. Author (M'76–SM'81–F'87) and the other authors may include biographies at the end of regular papers. Biographies are often not included in conference-related papers. The first paragraph may contain a place and/or date of birth (list place, then date). Next, the author's educational background is listed. The degrees should be listed with type of degree in what field, which institution, city, state or country, and year degree was earned. The author's major field of study should be lower-cased.

The second paragraph uses the pronoun of the person (he or she) and not the author's last name. It lists military and work experience, including summer and fellowship jobs. Job titles are capitalized. The current job must have a location; previous positions may be listed without one. Information concerning previous publications may be included. Try not to list more than three books or published articles. The format for listing publishers of a book within the biography is: title of book (city, state: publisher name, year) similar to a reference. Current and previous research interests ends the paragraph.

The third paragraph begins with the author's title and last name (e.g., Dr. Smith, Prof. Jones, Mr. Kajor, Ms. Hunter). If a photograph is provided, the biography will be indented around it. The photograph is placed at the top left of the biography. Personal hobbies will be deleted from the biography.