Terakreditasi SINTA Peringkat 2
Surat Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti No. 10/E/KPT/2019

Terbit online pada laman web jurnal: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 5 No. 5 (2021) 911 - 918 ISSN Media Elektronik: 2580-0760

Perbandingan Metode TF-ABS dan TF-IDF Pada Klasifikasi Teks Helpdesk Menggunakan K-Nearest Neighbor

Riza Adrianti Supono[1], Muhammad Azis Suprayogi[2]
[1,2] Manajemen Sistem Informasi, Program Pasca Sarjana, Universitas Gunadarma
[1]adrianti@staff.gunadarma.ac.id, [2]azissuprayogi.gunadarma@gmail.com*

Abstract

Distribution of tickets to the destination unit is a very important function in the helpdesk application, but the process of distributing tickets manually by admin officers has drawbacks, namely ticket distribution errors can occur and increase ticket completion time if the number of tickets is large. Helpdesk text classification becomes important to automatically distribute tickets to the appropriate destination units in a short time. This study was conducted to compare the performance of helpdesk text classification at the Directorate General of State Assets of the Ministry of Finance using the K-Nearest Neighbor (KNN) method with the TF-ABS and TF-IDF weighting methods. The research was conducted by collecting complaint documents, preprocessing, word weighting, feature reduction, classification, and testing. Classification using KNN with parameters n_neighbor (k) namely k=1, k=3, k=5, k=7, k=9, k=11, k=13, k=15, k=17, and k=19 to classify 10,537 helpdesk texts into 8 categories. The test uses a confusion matrix based on the accuracy value and score-f1. The test results show that the TF-ABS weighting method is better than TF-IDF with the highest accuracy value of 90.04% at 15% and k=3.

Keywords: *helpdesk, term weighting, text classification, tf-abs, tf-idf*

Abstrak

Penyaluran tiket ke unit tujuan merupakan fungsi yang sangat penting dalam aplikasi *helpdesk*, tetapi proses penyaluran tiket secara manual oleh petugas admin memiliki kekurangan yaitu dapat terjadi kesalahan penyaluran tiket dan meningkatkan waktu penyelesaian tiket apabila jumlah tiket banyak. Klasifikasi teks *helpdesk* menjadi penting untuk menyalurkan tiket secara otomatis ke unit tujuan yang sesuai dalam waktu yang singkat. Penelitian ini dilakukan untuk membandingkan performa klasifikasi teks *helpdesk* pada Direktorat Jenderal Kekayaan Negara Kementerian Keuangan menggunakan metode *K-Nearest Neighbor (KNN)* dengan metode pembobotan TF-ABS dan TF-IDF. Penelitian dilakukan dengan cara mengumpulkan dokumen pengaduan, *preprocessing*, pembobotan kata, reduksi fitur, klasifikasi, dan pengujian. Klasifikasi menggunakan KNN dengan parameter *n_neighbor* (k) yaitu k=1, k=3, k=5, k=7, k=9, k=11, k=13, k=15, k=17, dan k=19 untuk mengklasifikasikan teks *helpdesk* sebanyak 10.537 dokumen ke dalam 8 kategori. Pengujian menggunakan confusion matrix berdasarkan nilai akurasi dan score-f1. Hasil pengujian menunjukkan bahwa metode pembobotan TF-ABS lebih baik dari TF-IDF dengan nilai akurasi tertinggi 90,04% pada pada jumlah fitur 15% dan k=3.

Kata kunci: helpdesk, klasifikasi teks, pembobotan kata, tf-abs, tf-idf

1. Pendahuluan

Helpdesk merupakan sistem aplikasi yang berguna bagi pengguna nya untuk mendapatkan informasi tentang layanan proses bisnis pada sebuah organisasi. Direktorat Jenderal Kekayaan Negara (DJKN) Kementerian Keuangan sebagai salah satu instansi pemerintah yang menggunakan *helpdesk* untuk memudahkan pemangku kepentingan baik pegawai DJKN maupun pihak luar untuk mendapatkan informasi terkait layanan DJKN. Alur layanan *helpdesk* DJKN dapat dijelaskan secara ringkas yaitu dimulai dari pengguna layanan membuat tiket *helpdesk*, kemudian tim PPID atau *admin* mendistribusikan tiket tersebut secara manual ke direktorat atau unit lain sesuai dengan kewenangannya untuk ditindaklanjuti dan diselesaikan. *Helpdesk* DJKN membagi tujuan tiket ke dalam 8 (delapan) direktorat yaitu Sekretariat Direktorat Jenderal, Direktorat Kekayaan Negara Dipisahkan, Direktorat Barang Milik Negara, Direktorat Piutang Negara dan Kekayaan Negara Lain-lain, Direktorat Pengelolaan Kekayaan Negara dan Sistem Informasi, Direktorat Penilaian, Direktorat Hukum dan Hubungan Masyarakat, dan Direktorat Lelang.

Diterima Redaksi: 11-08-2021 | Selesai Revisi: 22-09-2021 | Diterbitkan Online: 25-10-2021

Selama dua tahun, mulai tahun 2019 sampai dengan tahun 2020, jumlah tiket helpdesk DJKN yang masuk dan selesai ditindaklanjuti berjumlah 10.537 tiket. Adapun berdasarkan *Standard Operational Procedure (SOP)*, waktu maksimal untuk ekskalasi tiket ke unit tujuan adalah 2 x 24 jam. Jumlah tim *admin* yang terbatas dan angka tiket yang cukup banyak diperlukan ketelitian tim *admin* dalam mengklasifikasikan tujuan tiket. Hal ini menyebabkan proses penyaluran tiket tersebut menjadi lambat dan dapat mengalami kesalahan unit tujuan. Oleh karena itu diperlukan metode klasifikasi yang dapat melakukan klasifikasi terhadap tiket *helpdesk* yang masuk untuk menentukan tujuan unit tiket sesuai dengan kewenangannya secara tepat sehingga diharapkan dapat menggantikan proses ekskalasi tiket ke unit tujuan secara manual.

Text Mining merupakan metode yang umum digunakan untuk klasifikasi teks. *Text Mining* merupakan metode untuk mengkonversi teks yang bentuknya tidak terstruktur menjadi data teks semi-terstruktur, untuk menemukan pola diantara teks tersebut dan menyelesaikan masalah [1]. Tujuan klasifikasi teks adalah untuk melakukan klasifiksai secara otomatis terhadap dokumen berbentuk teks berdasarkan kategori yang sebelumnya sudah dilakukan training. Adapun beberapa contoh aplikasi terkait klasifikasi teks yaitu analisis sentimen, klasifikasi dokumen, klasifikasi *email spam* dan peringkasan dokumen [2].

Beberapa algoritma yang umum diterapkan pada *text mining* khususnya untuk klasifikasi adalah algoritma *K-Nearest Neighbor (KNN)*, karena bertujuan untuk mengklasifikasikan objek berdasarkan atribut dan data latih [3]. Algoritma KNN merupakan salah satu metode klasifikasi yang popular dengan hasil yang akurat dan mudah dipahami [4].

Beberapa peneliti pernah melakukan penelitian mengenai klasifikasi dokumen teks, yaitu klasifikasi jenis laporan masyarakat dengan *K-Nearest Neighbour algorithm* [4]. Pada penelitian ini algoritma *K-Nearest Neighbour* menghasilkan akurasi yang baik pada klasifikasi data laporan masyarakat ke dalam tiga kategori yaitu pengaduan, permintaan informasi dan aspirasi, berdasarkan evaluasi serta validasi dengan *Confusion Matrix* ditemukan bahwa akurasi tertinggi adalah 82% dengan parameter k=11.

Penelitian selanjutnya yaitu Analisa perbandingan metode *Naïve Bayes Classifier* dan *K-Nearest Neighbour* terhadap klasifikasi data forum diskusi mahasiswa menjadi kategori judul topik berdasarkan isi materi [5]. Hasil penelitian tersebut menunjukkan bahwa metode NBC dan KNN dapat digunakan untuk klasifikasi data forum, pengukuran terhadap efektifitas klasifikasi terhadap 15 data uji diperoleh nilai 80% untuk metode KNN dan nilai 73% untuk metode NBC dengan menggunakan *Confusion Matrix*, sehingga disimpulkan bahwa metode KNN lebih baik dibandingkan dengan metode NBC.

Penelitian berikutnya, analisis sentimen opini publik berita kebakaran hutan[6] bertujuan mendapatkan nilai komparasi akurasi algoritma SVM dan KNN, untuk akurasi KNN tanpa seleksi fitur *Particle Swarm Optimation* sebesar 85% lebih baik dari pada SVM tanpa seleksi fitur *Particle Swarm Optimation* dengan nilai akurasi sebesar 80,83%. Penelitian berikutnya yaitu komparasi algoritma klasifikasi pada analisis review hotel[7] dimana, hasil dari komparasi algoritma klasifikasi antara *Naïve bayes (NB)*, *Support Vector Machine (SVM)*, dan *K-Nearest Neighbor (KNN)* didapatkan bahwa KNN memiliki hasil terbaik dengan akurasi 75% dan nilai AUC 0,500.

Penelitian berikutnya[8] mendemonstrasikan kecepatan proses *training* pada klasifikasi tiket pada *issue tracking system* yang menunjukkan bahwa algoritma *K-Nearest Neighbor* lebih cepat dari pada *Support Vector Machine.*

Proses pembobotan kata (*term weighting*) dilakukan untuk mengubah dokumen teks dari bentuk yang tidak terstruktur menjadi bentuk yang terstruktur serta untuk menentukan nilai kontribusi suatu kata pada dokumen terhadap kategori klasifikasi tertentu. Metode pembobotan kata bertujuan untuk memberi pengaruh pada performa klasifikasi dokumen [9]. Terdapat dua jenis metode pembobotan kata yaitu *Unsupervised Term Weighting (Traditional Term Weighting)* dan *Supervised Term Weighting*[10]. Contoh metode *unsupervised term weighting* antara lain *Term Frequency (TF)* dan *Term Frequency-Inverse Document Frequency (TF-IDF)* dan lain-lain, sedangkan metode *supervised term weighting* merupakan metode dimana informasi tentang keanggotaan dokumen pelatihan ke dalam kategori diperhitungkan dalam perhitungan bobot suatu kata [11]. Contoh metode *supervised term weighting* antara lain *Term Frequency-Relevance Frequency (TF-RF)*, *Term Frequency-Absolute (TF-ABS)*, dan *Term Frequency-Chisquare (TF-CHI^2)*.

Penelitian mengenai perbandingan pembobotan kata pada proses klasifikasi yang telah dilakukan yaitu kategorisasi berita menggunakan metode pembobotan TF-ABS dan TF-CHI dan algoritma *Support Vector Machine (SVM)* pada berita berbahasa inggris [9] dengan hasil bahwa kategorisasi berita tanpa proses *stemming* menggunakan metode pembobotan TF-ABS menghasilkan akurasi 95,74%, sedangkan menggunakan metode pembobotan TF-CHI menghasilkan akurasi 95,87% sehingga disimpulkan bahwa metode pembobotan TF-ABS dan TF-CHI tidak memberikan beda yang signifikan dalam performa serta menunjukkan bahwa kedua metode tersebut memiliki performa yang baik dalam kategorisasi berita.

Penelitian lain dilakukan pada klasifikasi topik berita dalam bahasa Indonesia menggunakan *Decision Tree*

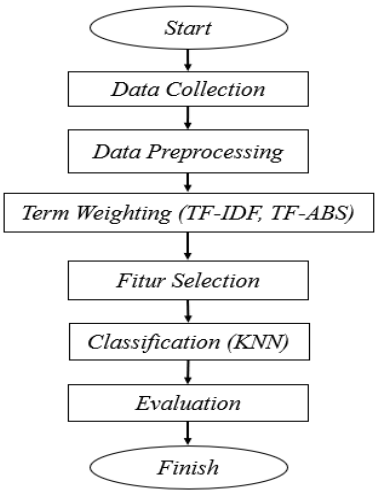
DOI: https://doi.org/10.29207/resti.v5i5.3403

dan pembobotan kata TF, TF-IDF, TF-RF, TF-ABS, TF-CHI^2[9] dengan kesimpulan TF-ABS merupakan Teknik pembobotan terbesar dengan nilai akurasi 82,22%. Namun penelitian lainnya yaitu perbandingan fitur pembobotan TF-IDF, TF-CHI, TF-RF dan TF-OR pada *Support Vector Machine* untuk *sentiment analysis* pada Jakarta BRT[9] menghasilkan TF-IDF sebagai fitur pembobotan yang memiliki performa terbaik dibandingkan fitur pembobotan lainnya dengan nilai akurasi 79,3%, nilai *precission* 83,2%, nilai *recall* 83,6% dan nilai *F1 score* 82,2%.

Berdasarkan beberapa penelitian tersebut, maka penelitian ini dilakukan untuk membandingkan performa klasifikasi teks *helpdesk* menggunakan metode pembobotan kata TF-ABS dan TF-IDF. Perbandingan performa klasifikasi dilakukan dengan cara menghitung nilai akurasi dan *score-f1* metode TF-ABS dan TF-IDF menggunakan *confusion matrix*. Adapun algoritma yang digunakan untuk klasifikasi adalah *K-Nearest Neighbor*.

2. Metode Penelitian

Kerangka kerja secara umum dapat dilihat pada Gambar 1, proses terdiri dari 5 tahapan yaitu proses pengumpulan data (*Data Collection*), praproses teks (*Data Preprocessing*), proses pembobotan kata (*Term Weighting*) metode TF-IDF dan TF-ABS, seleksi fitur (Fitur Selection), proses klasifikasi (*Classification*) menggunakan algoritma *K-Nearest Neighbor* dan proses evaluasi (*Evaluation*).



Gambar 1. Kerangka kerja penelitian

2.1. Pengumpulan Data dan *Preprocessing*

Data yang diambil adalah berasal dari tiket helpdesk yang dibuat pada tahun 2019 sampai dengan tahun 2020 yang berasal dari jalur *call center, email idcc* dan *website helpdesk idcc* berjumlah 10.537 tiket. Atribut yang dibutuhkan untuk klasifikasi teks tiket *helpdesk* adalah uraian isi tiket dan kategori tujuan tiket. Adapun kategori tujuan tiket *helpdesk* berjumlah 8 yaitu Sekretariat Direktorat Jenderal, Direktorat Barang Milik Negara, Direktorat Kekayaan Negara Dipisahkan, Direktorat Piutang Negara dan Kekayaan Negara Lain-lain, Direktorat Pengelolaan Kekayaan Negara dan Sistem Informasi, Direktorat Penilaian, Direktorat Lelang, dan Direktorat Hukum dan Hubungan Masyarakat. Tabel 1 memperlihatkan contoh data yang digunakan untuk penelitian.

Tabel 1. Contoh Data Teks *Helpdesk*

Uraian insiden	Kategori
Mohon informasi mengenai surat dengan nomor : 042/SRC/HLO/X/2018, pengirim Hadi Purwanto, Perihal Permohonan Informasi Status Tanah, apakah sudah diterima atau belum, dan sudah sampai mana progresnya?	Set. Ditjen
alas hak BMN untuk kementerian/instansi pemerintah itu apa ya? apakah ada peraturan di DJKN terkait hal tersebut?	Dit. Bmn
bertemu pa Iwang KND	Dit. KND
Bertanya mengenai status surat permohonan penyelesaian hutang piutang yang dikirimkan ke Direktorat PKNSI.	Dit. PNKNL
Minta disambungkan ke bagian yang mengurus tukar menukar BMN.	Dit. PKNSI
Permohonan penilaian untuk PBB apakah bisa melalui DJKN?	Dit. Penilaian
Aanwijzing apakah bisa langsung datang di waktu yang sudah dijadwalkan?	Dit. Lelang
BERTEMU DENGAN STAFF HUHU KONSUL VALIDASI	Dit. Humas

Adapun jumlah dokumen pada masing-masing kelas tujuan teks *Helpdesk* dapat dilihat pada Tabel 2.

Tabel 2. Jumlah Dokumen Teks *Helpdesk*

Kelas	Jumlah
Set. Ditjen	1.462
Dit. Bmn	1.226
Dit. KND	50
Dit. PNKNL	111
Dit. PKNSI	5.145
Dit. Penilaian	41
Dit. Lelang	2.279
Dit. Humas	223
Jumlah	10.537

Preprocessing data dilakukan melalui beberapa tahapan, dimulai dari tahap *case folding* yaitu mengubah teks dokumen menjadi huruf kecil seluruhnya, kemudian tahap *tokenization* yaitu membersihkan teks dari tanda baca, spasi berulang, mengubah baris baru menjadi spasi, dan memisahkan kata per kata dari kalimatnya, *stopword* yaitu menghapus kata yang tidak dibutuhkan, dan tahap *stemming* yaitu menghilangkan imbuhan kata pada teks dokumen. Adapun prosesnya menggunakan *tools* berbasis Python berupa *library Pandas, NLTK, dan Sastrawi.* Proses *case folding* dilakukan menggunakan *method str.lower()* dari *library Pandas*, selanjutnya proses *tokenization* dilakukan menggunakan *method word_tokenize()* pada *library NLTK*, kemudian proses *stopword* dilakukan dengan cara menghapus kata-kata

DOI: https://doi.org/10.29207/resti.v5i5.3403

yang tidak diperlukan menggunakan kamus *stopword* Indonesia pada *module stopwords library nltk.corpus*, selanjutnya tahapan *stemming* dilakukan menggunakan *module Sastrawi.Stemmer*.

2.2. Pembobotan TF-ABS dan TF-IDF

Proses pembobotan dilakukan setelah *preprocessing*. Pembobotan pertama dengan menggunakan metode *Term Frequency – Inverse Document Frequency (TF-IDF)* yaitu menghitung bobot *term* pada sebuah dokumen berdasarkan seringnya kata tersebut muncul dimana bobot tersebut mengindikasikan pentingnya sebuah *term* terhadap dokumen, semakin banyak *term* tersebut muncul pada dokumen maka semakin tinggi nilai *term* tersebut[12]. Teknik TF-IDF juga mengeliminasi *term* yang bersifat sangat umum dan mengekstrak *term* yang memiliki relevansi yang tinggi dari *corpus*[13]. Tahap pertama menentukan nilai *Term Frequency (TF)* yaitu jumlah *term* yang terdapat pada setiap dokumen, tahap selanjutnya menentukan nilai *Inverse Document Frequency (IDF)* yang berfungsi untuk mengurangi bobot *term* yang jumlah kemunculannya banyak di seluruh dokumen menggunakan Persamaan (1). Selanjutnya menentukan bobot *term* dengan cara mengalikan nilai TF dengan IDF menggunakan Persamaan (2).

$$idf_i = \log\left(\frac{N}{df_i}\right) \quad (1)$$

$$w_{i,j} = tf_{i,j} \times idf_i \quad (2)$$

dimana $w_{i,j}$ adalah bobot *term i* pada dokumen *j*, $tf_{i,j}$ adalah jumlah kemunculan *term i* pada dokumen *j*, idf_i adalah *inverse* df_i, df_i adalah banyaknya dokumen yang memuat *term* ke-*i*. N adalah jumlah dokumen. Contoh perhitungan pembobotan TF-IDF dapat dilihat pada Tabel 3.

Tabel 3. Contoh proses pembobotan TF-IDF

Uraian	Output
dokumen j	lelang kpknl tegal lolos serta lelang uang jaminan kembali
hasil *preprocessing*	
term i	lelang
TF	2
idf_i	log (10537/1862) = 0.752737303405706 2
$w_{i,j}$	2 x 0.7527373034057062 = 1.505474607

Pembobotan kedua menggunakan metode *Term Frequency – Absolute (TF-ABS)*. TF-ABS merupakan pengukuran kemungkinan suatu term t_j yang ada dalam dokumen dengan kategori c_i dibagi dengan kemungkinan term t_j yang tidak ada dalam dokumen dengan kategori tersebut dan menggunakan basis log e yang dikenal dengan logit. Perhitungan ABS atau ABSL atau abs-logit dengan *term* t_j dan category c_k dapat dilihat pada Persamaan (3).

$$ABS(t_j, c_k) = \left|\ln\left(\frac{(n_{kj}+0.5)(n_{\bar{k}\bar{j}}+0.5)}{(n_{\bar{k}j}+0.5)(n_{k\bar{j}}+0.5)}\right)\right| \quad (3)$$

dengan ABS adalah bobot kata, n_{kj} adalah jumlah dokumen pada kategori c_k dan mengandung *term* t_j, $n_{\bar{k}j}$ adalah jumlah dokumen tidak pada kategori c_k dan mengandung *term* t_j, $n_{k\bar{j}}$ adalah jumlah dokumen pada kategori c_k dan tidak mengandung term t_j, $n_{\bar{k}\bar{j}}$ adalah jumlah dokumen tidak pada kategori c_k dan tidak mengandung *term* t_j, t_j adalah *term* t_j, c_k adalah kategori c_k. Persamaan ABS merupakan salah satu metode seleksi kata yang umum digunakan. Persamaan ABS merupakan pengembangan dari metode seleksi *odds ratio*. Mengingat bahwa fokus pada kategorisasi teks adalah untuk kata yang terdistribusi secara berbeda pada kategori c_k dan $c_{\bar{k}}$, tidak penting apakah term tersebut lebih lazim pada kategori c_k dan $c_{\bar{k}}$, semakin memadai untuk tujuan ini adalah penggunaan nilai absolut dari *logit* sebagaimana persamaan (3) [14].

Proses selanjutnya adalah seleksi fitur (*feature selection*). Seleksi fitur merupakan strategi yang terbukti efektif dan efisien dalam menyiapkan data berdimensi tinggi untuk permasalahan *data mining* dan *machine learning* dengan tujuan untuk membangun model yang lebih sederhana, menyiapkan data yang lebih bersih, lebih mudah dipahami, serta meningkatkan kinerja data mining[15]. Metode *filter* sebagai salah satu metode seleksi fitur [15] dipilih pada penelitian ini karena metode filter tidak bergantung pada algoritma pembelajaran apapun karena mengandalkan pada karakteristik data tertentu untuk menilai bobot atau kepentingan suatu fitur. Metode filter dilakukan dengan dua langkah, pertama nilai bobot kepentingan suatu fitur atau kata diberi peringkat berdasarkan skor kata yang dihitung dengan mtode TF-ABS dan TF-IDF. Langkah kedua adalah menyaring kata dengan cara membuang kata-kata yang memiliki nilai bobot kepentingan yang rendah dan memilih fitur sisanya dengan nilai bobot kepentingan yang tinggi. Jumlah fitur yang dipilih ditentukan sebesar 5%, 10%, 15%, 20%, 25%, dan 30% dari total jumlah dokumen. Selanjutnya fitur hasil seleksi tersebut akan diuji untuk mendapatkan hasil klasifikasi yang terbaik.

2.3. Klasifikasi dan Evaluasi

Proses *split data* dilakukan sebelum melakukan klasifikasi dengan cara membagi *dataset* menjadi data training dan data testing. Data training digunakan sebagai data latih model, sedangkan data testing digunakan sebagai data uji model yang dilatih sebelumnya. Proses split data dokumen tersebut memanfaatkan fungsi train_test_split() pada modul *sklearn.model_selection..Dataset helpdesk* DJKN seluruhnya berjumlah 10.537 dokumen dibagi ke dalam data *training* dan data *testing* dengan perbandingan 90:10 sehingga jumlah data *training* adalah 9.483 dokumen, sedangkan jumlah data testing adalah 1.054

DOI: https://doi.org/10.29207/resti.v5i5.3403

dokumen. Tabel 4 memperlihatkan hasil distribusi data setelah *split data*.

Tabel 4. Distribusi dataset setelah *split data*

Kelas	*Data training*	*Data testing*
Set. Ditjen	1317	145
Dit. Humas	205	18
Dit. PKNSI	4629	516
Dit. KND	42	8
Dit. BMN	1097	129
Dit. Penilaian	35	6
Dit. PNKNL	99	12
Dit. Lelang	2059	220
Jumlah	9483	1054

Proses klasifikasi menggunakan algoritma *K-Nearest Neighbor* memanfaatkan fungsi *KNeighborsClassifier()* pada modul klasifikasi *sklearn.neighbors.* Model algoritma *K-Nearest Neighbor* dibuat menggunakan ukuran jarak *Euclidean Distance* dan nilai k yang umum digunakan yaitu k=1,3,5,7, 11,13,15,17, dan 19 untuk kemudian dipilih nilai k dengan performa terbaik. *Euclidean distance* antara dua data X1 dan X2 berarti bahwa untuk setiap atribut numerik akan diambil selisih nilai dari atribut tersebut pada data X1 dan X2, selisih tersebut kemudian dikuadratkan dan diakumulasikan, kemudian niai akar kuadrat diambil dari akumulasi selisih jarak[16]. *Euclidean distance* dihitung dengan Persamaan (4)

$$dist(X1, X2) = \sqrt{\sum_{i=1}^{n}(x_{1i} - x_{2i})^2} \quad (4)$$

Tahapan selanjutnya adalah evaluasi klasifikasi menggunakan pengukuran nilai akurasi dan *score-f1* dalam satuan persen berdasarkan rumusan menggunakan *Confusion Matrix*. *Confusion matrix* merupakan alat yang berfungsi menganalisis seberapa baik suatu *classifier* dalam melakukan klasifikasi[16].

Tabel 5. *Confusion Matrix*

Terprediksi	Aktual	
	Positif	Negatif
Positif	*True Positive (TP)*	*False Negative (FN)*
Negatif	*False Positive (FP)*	*True Negative (TN)*

True Positive (TP) adalah data yang diklasifikasikan berkaitan dengan kategori yang benar, *False Positive (FP)* adalah data yang diklasifikasikan berkaitan dengan kategori yang salah, *True Negative (TN)* adalah data yang diklasifikasikan tidak berkaitan dengan kategori yang benar, *False Negative (FN)* adalah data yang diklasifikasikan tidak berkaitan dengan kategori yang salah[17]. Terdapat beberapa pengukuran untuk menghitung performa klasifikasi yaitu akurasi, *precision*, *recall*, dan *score-f1*. Akurasi adalah jumlah proporsi prediksi yang benar[18], didefinisikan dalam persamaan (5)

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \quad (5)$$

Precision adala tingkat ketepatan antara informasi yang diinginkan user dengan jawaban yang diberikan sistem. *Precision* didefinisikan dalam Persamaan (6)

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

Recall adalah tingkat keberhasilan sistem dalam menemukan informasi. *Recall* didefinisikan dalam Persamaan (7)

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

Adapun *score-f1 (f-measure)* merupakan perhitungan evaluasi untuk menemukan kembali informasi yang mengkombinasikan *precision* dan *recall*. *Score-f1* didefinisikan dalam Persamaan (8)

$$score - f1 = \frac{2\ x\ Precision\ x\ Recall}{Precision+Recall} \quad (8)$$

3. Hasil dan Pembahasan

Hasil tahapan *preprocessing* yang dilakukan, yaitu *case folding*, *tokenization*, *stopword*, dan *stemming*. Contoh *preprocessing* pada teks helpdesk dapat dilihat pada Tabel 6.

Tabel 6. Contoh Teks Hasil *Preprocessing*

Tahapan	Hasil
Raw Data	ikut lelang di KPKNL TEGAL tapi tidak lolos jadi peserta lelang, apakah uang jaminan bisa dikembalikan?
Case Folding	ikut lelang di kpknl tegal tapi tidak lolos jadi peserta lelang, apakah uang jaminan bisa dikembalikan?
Tokenization	ikut lelang di kpknl tegal tapi tidak lolos jadi peserta lelang apakah uang jaminan bisa dikembalikan
Stopword	lelang kpknl tegal lolos peserta lelang uang jaminan dikembalikan
Stemming	lelang kpknl tegal lolos serta lelang uang jaminan kembali

Selanjutnya menghitung nilai bobot atau kepentingan kata menggunakan metode TF-IDF dimulai dengan menghitung jumlah kata pada pada dokumen atau nilai TF (*Term Frequency*) kemudian menghitung nilai IDF (*Inverse Document Frequency*) selanjutnya mengalikan nilai TF dengan IDF. Hasil perkalian tersebut akan digunakan untuk perhitungan klasifikasi KNN. Contoh nilai TF-IDF dapat dilihat pada Tabel 7.

Tabel 7. Contoh Perhitungan Nilai TF-IDF

Term	TF	IDF	TF-IDF=TF x IDF
lelang	2	0.752737303405706	1.5054746068114100
kpknl	1	0.6027612315612721	0.6027612315612721
tegal	1	2.743963379098201	2.743963379098201
lolos	1	3.177618940036773	3.177618940036773
serta	1	1.6744121170028694	1.6744121170028694
uang	1	1.3534000994849178	1.3534000994849178
jamin	1	1.53699555356945	1.53699555356945
kembali	1	1.5398433964422762	1.5398433964422762

DOI: https://doi.org/10.29207/resti.v5i5.3403

Berdasarkan hasil seleksi fitur yang dilakukan terhadap nilai kepentingan kata metode TF-IDF, kata-kata tersebut diurutkan mulai dari nilai kepentingan tertinggi hingga terendah kemudian diambil kata-kata yang memiliki nilai kepentingan tertinggi berjumlah 5%, 10%, 15%, 20%, 25%, dan 30% dari total jumlah dokumen.

Tabel 8. Contoh 20 kata bobot TF-IDF tertinggi

Term	Bobot
where	109.18694
set	99.6319
kpkp	95.122709
userid	84.477057
admuser	78.155427
fullname	78.155427
biad	75.613935
pratama	70.518574
kpp	69.975627
update	62.115799
permanenlpk	60.340755
sisa	60.100873
kp	59.855108
psbdt	53.663837
negara	52.444242
rev	49.079841
tanggal	48.903639
lampung	45.647086
paksa	44.468541
iprvbangunan	44.249887

Selanjutnya menghitung bobot TF-ABS dengan cara menghitung nilai TF dan nilai ABS menggunakan persamaan (3). Contoh hasil perhitungan tersebut dapat dilihat pada Tabel 9.

Tabel 9. Contoh Perhitungan Nilai TF-ABS

Term	TF	ABS	TF-ABS=TF x ABS
lelang	2	3.25703359287938	6.514067185758753
kpknl	1	1.0228030380963302	1.0228030380963302
tegal	1	0.3921471269703963	0.3921471269703963
lolos	1	1.6138147119640753	1.6138147119640753
serta	1	1.4521643285802575	1.4521643285802575
uang	1	1.23650377929861	1.23650377929861
jamin	1	1.9729247145771094	1.9729247145771094
kembali	1	2.0334233120840373	2.0334233120840373

Hasil seleksi fitur berdasarkan nilai TF-ABS diurutkan dari nilai tertinggi dan mengambil jumlah fitur yang ditentukan sebanyak 5%, 10%, 15%, 20%, 25%, dan 30%.

Klasifikasi menggunakan data sebanyak 10.537 dokumen yang merupakan hasil tahap *preprocessing* dan tahap pembobotan TF-IDF serta TF-ABS. Klasifikasi dilakukan menggunakan algoritma KNN dengan variasi nilai k=1, k=3, k=5, k=7, k=9, k=11, k=13, k=15, k=17, dan k=19 serta terhadap variasi jumlah fitur yang dihasilkan dari proses seleksi fitur sebanyak 5%, 10%, 15%, 20%, 25%, dan 30%. Pengukuran dilakukan terhadap nilai akurasi dan *score-f1* untuk masing-masing kategori yaitu Set.Ditjen, Dit. Bmn, Dit.KND, Dit.PNKNL, Dit.PKNSI, Dit.Penilaian, Dit.Lelang, Dit.Humas. Perhitungan akurasi berdasarkan *confusion matrix* untuk TF-IDF dan TF-ABS pada jumlah fitur 15% dan nilai k=3 dapat dilihat pada Tabel 11 dan Tabel 12.

Tabel 10. Contoh 20 kata bobot TF-ABS tertinggi

Term	bobot
biad	100.675697867951
where	96.5717769993511
lelang	75.5304728339422
lpk	65.7740212492694
tiket	65.5076233759041
sisa	57.8476435624348
investasi	57.142386936058
paksa	55.421125981491
nip	49.9883191031875
hutang	42.2485788028111
dukcapil	40.0491911503731
kpp	39.7350485574961
nilai	38.9637984652774
tanggal	38.8712366799404
pratama	38.6109523529609
kpkp	37.960523643658
wkn	37.9068180932011
set	37.0941678593576
ip	36.8144113568725
ujl	36.0973058660956

Tabel 11. Confusion Matrix TF-IDF

	Actual							
Dit.Setditjen	86	5	50	1	20	0	4	31
Dit.Humas	1	2	7	0	2	0	0	0
Dit.PKNSI	22	6	409	3	31	2	4	6
Dit.KND	0	0	1	1	1	0	0	0
Dit.BMN	5	3	16	1	69	1	0	0
Dit.Penilaian	0	0	0	0	0	2	0	0
Dit.PNKNL	2	0	4	2	1	0	4	0
Dit.Lelang	29	2	29	0	5	1	0	183

Akurasi = (86+2+409+1+69+2+4+183)/1054 x 100%
= 71,73%

Tabel 12. Confusion Matrix TF-ABS

	Actual							
Dit.Setditjen	130	7	27	0	7	0	2	0
Dit.Humas	3	5	3	0	0	0	0	0
Dit.PKNSI	6	5	473	0	9	3	1	5
Dit.KND	0	0	0	4	0	0	0	0
Dit.BMN	4	1	1	2	112	0	1	1
Dit.Penilaian	1	0	1	0	0	3	0	0
Dit.PNKNL	0	0	1	2	1	0	8	0
Dit.Lelang	1	0	10	0	0	0	0	214

Akurasi = (130+5+473+4+112+3+8+214)/1054 x 100%
= 90,04%

Tabel 13 dan Tabel 14 menunjukkan bahwa TF-ABS memiliki akurasi mulai dari 84,54% sampai dengan 90,04%, sedangkan TF-IDF memiliki akurasi mulai dari 66,89% sampai dengan 71,82%. Akurasi tertinggi diperoleh metode TF-ABS sebesar 90,04%. Dari Gambar 2 dapat diketahui bahwa akurasi metode TF-ABS lebih tinggi dari TF-IDF untuk seluruh nilai k dan seluruh penggunaan jumlah fitur. Gambar 2 juga menunjukkan bahwa penggunaan jumlah fitur berpengaruh pada akurasi yang dihasilkan. Nilai akurasi

DOI: https://doi.org/10.29207/resti.v5i5.3403

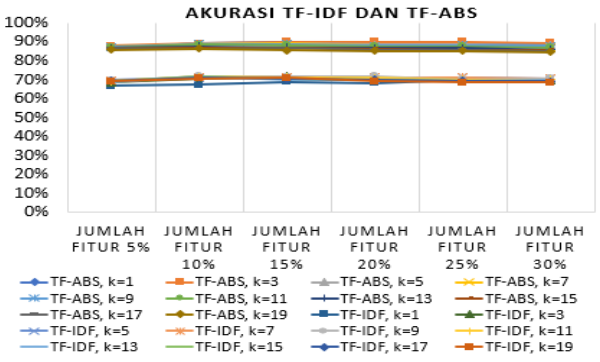
naik jika jumlah fitur ditambah dari 5% sampai dengan 15% diperoleh akurasi tertinggi kemudian akurasi mengalami penurunan setelah jumlah fitur ditambah menjadi 20%, 25%, dan 30%. Percobaan klasifikasi menunjukkan bahwa perubahan nilai k=1, k=3, k=5, k=7, k=9, k=11, k=13, k=15, k=17, dan k=19 juga berpengaruh pada akurasi. Akurasi tertinggi diperoleh menggunakan nilai k=3. Gambar 2 menunjukkan bahwa akurasi meningkat dari dari k=1 sampai k=3, tetapi kemudian akurasi menurun setelah nilai k ditambah sampai k=19 diperoleh akurasi terendah.

Tabel 13. Akurasi TF-IDF (%)

n_ neighbor	Jumlah fitur					
	5%	10%	15%	20%	25%	30%
k=1	66.89	67.46	68.88	68.31	69.83	69.54
k=3	68.69	70.68	71.73	70.40	70.11	69.83
k=5	69.83	70.97	71.35	70.40	71.16	70.68
k=7	69.54	71.35	71.92	71.25	71.16	70.59
k=9	69.54	71.82	72.11	71.82	69.92	70.59
k=11	69.26	71.54	71.06	70.87	70.02	70.11
k=13	68.98	71.54	70.30	70.02	69.07	69.73
k=15	69.45	71.63	70.78	69.83	69.45	69.17
k=17	69.45	70.97	70.87	70.11	69.45	69.17
k=19	69.26	70.78	70.78	69.26	68.41	68.69

Tabel 14. Akurasi TF-ABS (%)

n neighbor	Jumlah fitur					
	5%	10%	15%	20%	25%	30%
k=1	86.81	88.99	89.56	89.66	89.28	88.52
k=3	87.95	89.09	90.04	89.94	89.85	89.37
k=5	87.00	89.09	88.99	88.43	88.52	88.24
k=7	87.19	88.71	88.99	87.95	88.14	87.38
k=9	87.38	88.71	88.80	87.95	88.33	87.86
k=11	86.91	88.43	88.05	87.67	87.67	87.00
k=13	86.62	87.57	87.10	86.62	86.53	85.77
k=15	86.62	86.62	86.53	86.24	85.86	85.67
k=17	86.72	86.43	86.62	85.86	85.67	85.10
k=19	85.77	86.34	85.58	85.10	85.29	84.54



Gambar 2. Perbandingan akurasi TF-IDF dan TF-ABS

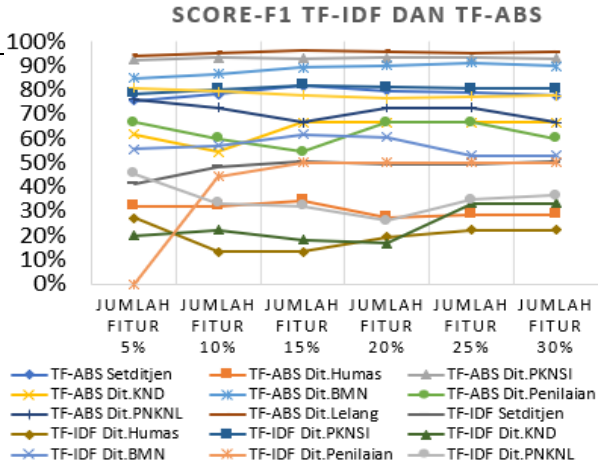
Selanjutnya berdasarkan nilai k=3 dilakukan pengukuran nilai *score-f1* untuk setiap kategori pada penggunaan setiap jumlah fitur baik untuk metode TF-IDF maupun TF-ABS.

Tabel 15. *Score-f1* TF-IDF (%)

n_ neighbor	Jumlah fitur					
	5%	10%	15%	20%	25%	30%
Setditjen	41.53	48.52	50.29	49.57	49.56	50.29
DitHumas	27.27	13.33	13.33	19.35	22.22	22.22
DitPKNSI	78.53	80.36	81.88	81.05	80.61	80.45
DitKND	20.00	22.22	18.18	16.67	33.33	33.33
DitBMN	55.65	57.14	61.61	60.43	52.85	52.85
DitPenilaian	00.00	44.44	50.00	50.00	50.00	50.00
DitPNKNL	45.45	33.33	32.00	26.09	34.78	36.36
DitLelang	80.72	79.31	78.04	76.32	77.38	77.64

Tabel 16. *Score-f1* TF-ABS (%)

n_ neighbor	Jumlah fitur					
	5%	10%	15%	20%	25%	30%
Setditjen	75.44	78.13	81.76	79.50	79.00	77.60
DitHumas	32.00	32.26	34.48	27.59	28.57	28.57
DitPKNSI	92.45	93.25	92.93	93.32	93.36	92.97
DitKND	61.54	54.55	66.67	66.67	66.67	66.67
DitBMN	84.92	86.49	89.24	89.96	91.13	89.68
DitPenilaian	66.67	60.00	54.55	66.67	66.67	60.00
DitPNKNL	76.19	72.73	66.67	72.73	72.73	66.67
DitLelang	94.09	95.30	96.18	95.75	95.07	95.50



Gambar 3. Perbandingan *score-f1* TF-IDF dan TF-ABS

Tabel 16 Menunjukkan bahwa nilai *score-f1* metode TF-ABS lebih tinggi dari pada TF-IDF pada Tabel 15 pada setiap kategori/kelas dan pada setiap penggunaan jumlah fitur. Pada TF-IDF, nilai *score-f1* terendah sebesar 0% pada kategori Dit.Penilaian dan jumlah fitur 5%, nilai score-f1 tertinggi sebesar 81,88% pada kategori Dit.PKNSI dan jumlah fitur 15%, sedangkan TF-ABS memiliki nilai *score-f1* terendah sebesar 27,59% pada kategori Dit.Humas dan jumlah fitur 20%, nilai *score-f1* tertinggi sebesar 96,18% pada kategori Dit.Lelang dan jumlah fitur 15%.

4. Kesimpulan

Berdasarkan percobaan klasifikasi teks *helpdesk* menggunakan KNN dengan metode pembobotan TF-IDF dan TF-ABS pada nilai k=1, k=3, k=5, k=7, k=9, k=11, k=13, k=15, k=17, dan k=19 serta menggunakan

DOI: https://doi.org/10.29207/resti.v5i5.3403

variasi jumlah fitur yaitu 5%, 10%, 15%, 20%, 25%, dan 30% diperoleh hasil akurasi tertinggi pada metode pembobotan TF-ABS sebesar 90,04% pada k=3 dan penggunaan jumlah fitur 15%. Hal ini menunjukkan bahwa metode pembobotan TF-ABS lebih baik daripada TF-IDF. Hal ini disebabkan karena metode TF-ABS memperhitungkan kategori dalam perhitungan pembobotannya dengan cara menghitung jumlah dokumen yang masuk kategori tertentu dan mengandung kata tertentu atau tidak mengandung kata tertentu, selain itu memperhitungkan jumlah dokumen yang tidak masuk ke dalam kategori tertentu dan mengandung kata tertentu atau tidak mengandung kata tertentu. Dengan kombinasi perhitungan seperti itu dapat menghitung bobot suatu kata dengan lebih tepat dibandingkan dengan perhitungan pembobotan TF-IDF yang tidak memperhitungkan kategori. Selain itu penggunaan jumlah fitur dan nilai *k* KNN berpengaruh terhadap akurasi klasifikasi. Akurasi meningkat dari penggunaan jumlah fitur 5% sampai jumlah fitur 15% diperoleh akurasi tertinggi, kemudian akurasi mengalami penurunan setelah jumlah fitur ditambah menjadi 20%, 25% dan 30%. Demikian hal nya dengan variasi n_neighbor, akurasi meningkat pada k=1 hingga k=3 diperoleh akurasi tertinggi, kemudian akurasi menurun setelah nilai k ditambah menjadi k=5 sampai dengan k=19.

Saran untuk penelitian selanjutnya adalah menambahkan metode untuk mengatasi data yang tidak seimbang antara kategori satu dengan yang lain, kemudian penelitian ini hanya menggunakan KNN untuk mengklasifikasi teks *helpdesk* dan menguji akurasi pembobotan TF-ABS, penggunaan metode klasifikasi yang lain sangat mungkin digunakan pada penelitian berikutnya. Dengan demikian diharakan dapat lebih meningkatkan akurasi klasifikasi terhadap teks *helpdesk*.

Daftar Rujukan

[1] R. Feldman and J. Sanger, *The Text Mining Handbook*. 2006.

[2] A. Kulkarni and A. Shivananda, *Natural Language Processing Recipes*. New York, USA: Apress, 2019.

[3] Okfalisa, I. Gazalba, Mustakim, and N. G. I. Reza, "Comparative analysis of k-nearest neighbor and modified k-nearest neighbor algorithm for data classification," *Proc. - 2017 2nd Int. Conf. Inf. Technol. Inf. Syst. Electr. Eng. ICITISEE 2017*, vol. 2018-Janua, pp. 294–298, 2018, doi: 10.1109/ICITISEE.2017.8285514.

[4] C. F. Suharno, M. A. Fauzi, and R. S. Perdana, "Klasifikasi Teks Bahasa Indonesia Pada Dokumen Pengaduan Sambat Online Menggunakan Metode K-Nearest Neighbors dan Chi-Square," *Syst. Inf. Syst. Informatics J.*, vol. 03, no. 01, pp. 25–32, 2017.

[5] A. Indriani, "Analisa Perbandingan Metode Naïve Bayes Classifier Dan K-Nearest Neighbor Terhadap Klasifikasi Data," *Sebatik*, vol. 24, no. 1, pp. 1–7, 2020, doi: 10.46984/sebatik.v24i1.909.

[6] L. A. Utami, "Melalui Komparasi Algoritma Support Vector Machine Dan K-Nearest Neighbor Berbasis Particle Swarm Optimization," vol. 13, no. 1, pp. 103–112, 2017.

[7] L. D. Utami, "Komparasi Algoritma Klasifikasi Pada Analisis Review Hotel," *J. Pilar Nusa Mandiri*, vol. 14, no. 2, p. 261, 2018, doi: 10.33480/pilar.v14i2.1023.

[8] M. ALTINTAġ and A. C. TANTUĞ, "Machine learning based software development," vol. 21, no. 3, pp. 33–44, 2014.

[9] M. A. Kurniawan, Y. Sibaroni, and K. L. Muslim, "Kategorisasi Berita Menggunakan Metode Pembobotan TF.ABS dan TF.CHI," *Indones. J. Comput.*, vol. 3, no. 2, p. 83, 2018, doi: 10.21108/indojc.2018.3.2.236.

[10] M. Lan, C. L. Tan, J. Su, and Y. Lu, "Supervised and traditional term weighting methods for automatic text categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 721–735, 2009, doi: 10.1109/TPAMI.2008.110.

[11] F. Debole and F. Sebastiani, "Supervised Term Weighting for Automated Text Categorization," in *Supervised Term Weighting for Automated Text Categorization*, 2003, no. December 2015, doi: 10.1145/952686.952688.

[12] N. G. Yudiarta, M. Sudarma, and W. G. Ariastina, "Penerapan Metode Clustering Text Mining Untuk Pengelompokan Berita Pada Unstructured Textual Data," *Maj. Ilm. Teknol. Elektro*, vol. 17, no. 3, p. 339, 2018, doi: 10.24843/mite.2018.v17i03.p06.

[13] P. Bafna, D. Pramod, and A. Vaidya, "Document clustering: TF-IDF approach," *Int. Conf. Electr. Electron. Optim. Tech. ICEEOT 2016*, no. November, pp. 61–66, 2016, doi: 10.1109/ICEEOT.2016.7754750.

[14] L. A. Matsunaga and N. F. F. Ebecken, "Two Novel Weighting for Text Categorization," in *Data Mining IX - Data Mining, Protection, Detection and other Security Technologies*, IX., A. Zanasi, D. Almorza Gomar, N. F. . Ebecken, and C. . Brebbia, Eds. Rio de Janeiro, Brazil: WITPRESS, 2008, pp. 105–114.

[15] J. Li *et al.*, "Feature selection: A data perspective," *ACM Comput. Surv.*, vol. 50, no. 6, 2017, doi: 10.1145/3136625.

[16] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques - third edition*. 2012.

[17] D. Yuliana and C. Supriyanto, "Klasifikasi Teks Pengaduan Masyarakat Dengan Menggunakan Algoritma Neural Network," *UPI YPTK J. KomTekInfo*, vol. 5, no. 3, pp. 92–116, 2019.

[18] L. A. Andika, P. A. N. Azizah, and R. Respatiwulan, "Analisis Sentimen Masyarakat terhadap Hasil Quick Count Pemilihan Presiden Indonesia 2019 pada Media Sosial Twitter Menggunakan Metode Naive Bayes Classifier," *Indones. J. Appl. Stat.*, vol. 2, no. 1, p. 34, 2019, doi: 10.13057/ijas.v2i1.29998.

DOI: https://doi.org/10.29207/resti.v5i5.3403