# Implementasi Metode TF-IDF dan K-Nearest Neighbor untuk Seleksi Pelamar Kerja

**Jofardho Adlinnas[1], Kemas Muslim Lhaksmana[2], Donni Richasdy[3]**

[1,2,3] Fakultas Informatika, Universitas Telkom, Bandung, Indonesia
Jl.Telekomunikasi No.1 Terusan Buah Batu Bndung
[1]jofardho@student.tekomuniversity.ac.id, [2] kemasmuslim@telkomuniversity.ac.id,
[3]donnirichasdy@telkomuniversity.ac.id

**Abstrak**

**Indonesia merupakan salah satu negara dengan jumlah penduduk terbesar didunia dan mengalami peningkatan disetiap tahunnya, maka dari itu jumlah tenaga kerja juga terus meningkat pada setiap tahunnya dari berbagai jenis tingkatan pendidikan. Perekrutan pegawai merupakan salah satu proses penting menyaring pelamar yang berkualifikasi dan memenuhi standar organisasi/perusahaan. Proses perekrutan pelamar kerja yang dengan jumlah yang banyak menjadikan salah satu faktor bagi perusahaan membutuhkan waktu dan biaya lebih pada proses penyeleksian. Salah satu cara untuk memudahkannya proses seleksi, dengan memberi label/skor pada hasil wawancara pelamar oleh expert/ahli. Untuk menyelesaikan masalah tersebut digunakannya metode *Term Frequency-Inverse Document Frequency* (TF-IDF) sebagai extraksi fitur dan metode *K-Nearest Neighbor* (KNN) dengan cosine similarity untuk menghitung jarak tetangga terdekat, sebagai klasifikasi terhadap teks hasil wawancara pelamar. Hasil dari proses ini menunjukkan bahwa KNN merupakan pendekatan yang cukup efektif karena tingkat akurasi KNN mampu menghasilkan keakuratan rata-rata mencapai 65.2%.**

**Kata kunci:** ***perekrutan pelamar kerja, klasifikasi teks, K Nearest-Neighbor, Cosine similarity***

***Abstract***

**Indonesia is one of the countries with the largest population in the world and has increased every year, therefore the number of workers also continues to increase every year from various types of education levels. Recruitment of employees is an important process of screening qualified applicants and meeting organizational / company standards. The recruitment process of job applicants with a large number makes one of the faktors for companies requiring more time and money in the selection process. One way to facilitate the selection process, by giving a label / score on the interview results of the applicant by the expert / expert. To solve this problem the term frequency-inverse document frequency (TF-IDF) method is used as a feature extraction and the K-Nearest Neighbor (KNN) method K-Nearest Neighbor (KNN) method with cosine similarity to calculate the distance to the nearest neighbor, as a classification of the text of the interview applicants. The results of this process show that KNN is a quite effective approach because the accuracy of KNN is able to produce an average accuracy of 65.2%.**

***Keywords*: *recruitment of job applicants, text classification, K Nearest-Neighbor, Cosine similarity***

## 1. Pendahuluan

### 1.1. Latar Belakang

Indonesia merupakan salah satu negara dengan jumlah penduduk terbesar di dunia dan mengalami peningkatan disetiap tahunnya, maka dari itu jumlah tenaga kerja juga terus meningkat pada setiap tahunnya dari berbagai jenis tingkatan pendidikan. Proses dalam pencarian pelamar kerja ini diawali dari perusahaan yang memberitahukan adanya lowongan pekerjaan yang dibutuhkan dari perusahaan tersebut baik melalui media cetak, ataupun media social. Bagi pelamar pekerja yang tertarik dengan posisi tersebut maka pelemar tersebut dapat mengirimkan *curriculum citae* (CV). Perusahaan pun akan melakukan penilaian terhadap cv yang telah dikirimkan, dan apabila pelamar tersebut lolos maka akan masuk ke tahap selanjutnya yaitu tahapan wawancara atau *interview*.

Banyak perusahaan yang menerima pelamar kerja yang mungkin tidak sesuai, yang disebabkan adanya ketidaksesuaian seperti bias dalam penseleksian (*confirmation bias, affinity bias*), perbedaan standar penilaian pewawancara, waktu yang lama terhadap penilaian hasil wawancara yang dikarenakan banyaknya jumlah pelamar kerja dan biaya yang dikeluarkan cukup mahal. Salah satunya contohnya PT.Telkom pada tahun 2018 terdapat 19.290 pelamar yang mencalonkan diri untuk berkerja di PT.Telkom dan hanya 138 orang yang diterima.

Pada penelitian ini dilakukannya penilaian atau klasifikasi teks (*teks classification*) hasil wawancara yang dilakukan oleh pelamar menggunakan pembelajaran mesin (*mechine learning*) banyak digunakan untuk megklasifikasi suatu teks [1]. Dengan menggunakan pembelajaran mesin ini diharapkan dapat membantu proses seleksi, mempercepat proses seleksi pelamar kerja. Oleh karena itu dibutuhkannya pembuatan suatu sistem penilaian atau klasifikasi teks dengan bantuan pembelajaran mesin untuk membuat proses seleksi lebih efisien.

Metode yang di pakai dalam penelitian ini adalah K-Nearest Neighbor(KNN). Beberapa keunggulan dari KNN adalah metode ini dapat secara alami memberikan probabilitas dan meluas ke masalah klasifikasi multi-label. Penelitian ini berfokus pada tahap wawancara, dimana data hasil wawancara akan dilakukan pelabelan. Pelabelan dilakukan oleh expert dengan memberikan tanda kelas atau label terhadap suatu teks.

Metode Cosine Similarity untuk mengukur/menghitung nilai kemiripan antar kalimat atau dokumen. Pada Cosine Similarity kalimat atau dokumen teks akan dianggap sebagai vector. Pada penelitian ini, Cosine Similarity digunakan untuk menghitung nilai jarak tetangga pada metode klasifikasi KNN. Semakin besar nilai jarak tetangga yang diperoleh maka nilai Cosine Similarity akan semakin tinggi.

Dataset yang diperoleh dalam penelitian ini didapatkan dari hasil wawancara. Data tersebut berjumlah sebanyak 54 data jawaban wawancara pelamar kerja yang terdiri dari *content* dan *core values*. Dataset ini memiliki 9 (sembilan) *core values* yaitu: *action, enthusiams, focus, imagine, integrity, smart, solid, speed* dan *totality*, dimana masing-masing core values terdiri dengan dua label/kelas yaitu 1 (layak) dan 2 (tidak layak). Untuk kategori sendiri merupakan permintaan dari pihak Telkom guna membuat prototype dan *prof of concept*, apakah memungkinkan adopsi *machine learning* dalam proses perecruitan pelamar kerja. Pada kasus penelitian ini dilakukan analisis hasil kinerja metode klasifikasi menggunakan KNN untuk mengetahui keakuratan akurasi dalam menangani klasifikasi untuk seleksi pelamar kerja. Tugas utama sistem diharapkan mampu mengidentifikasi dan mengklasifikasi data teks hasil wawancara berdasarkan jawaban kandidat pada saat wawancara. Serta dapat mempermudah proses seleksi pelamar kerja yang secara otomatis dapat menilai hasil wawancara atau jawaban kandidat sesuai dengan kriteria dari suatu perusahaan, mengoptimalkan proses rekrutmen karyawan dengan mengurangi waktu untuk merekrut, menghemat biaya, meminimalkan beberapa bias manusia tersembunyi yang mencegah kandidat minoritas dari mendapatkan evaluasi yang adil, dan membantu perusahaan meningkatkan kemampuan mencocokkan kandidat mereka. Dengan menggunakan kriteria tertentu, sistem yang dibangun dapat memprediksi potensi dari seorang pelamar dan dapat memberikan rekomendasi bagi divisi SDM terkait diterima atau tidaknya pelamar tersebut.Selain itu juga untuk meminimalisir waktu , biaya, dan mengurangi keterlibatan pihak ketiga dalam proses perecruitan karyawan.

### 1.2. Topik dan Batasannya

Permasalahan yang dibahas dalam tugas akhir ini adalah bagaimana merancang model klasifikasi seleksi pelamar kerja dengan data hasil wawancara menggunakan metode K Nearest-Neighbor(KNN). Bagaimana performa, kinerja dan akurasi dari sistem yang dibangun. Batasan masalah dalam penelitian tugas akhir ini adalah dataset yang digunakan dalam penelitian merupakan data teks wawancara pelamar pada PT. Telkom yang berjumlah 54 content, dengan 9 (sembilan) core values. Dataset disimpan dalam bentuk file Microsoft Excel Comma Separated Values (.csv). Dataset yang digunakan pada penelitian ini sangat sedikit dengan jumlah 54 data.

### 1.3. Tujuan

Tujuan dari Tugas Akhir ini untuk melakukan klasifikasi data teks hasil wawancara dari pelamar dengan menggunakan metode *term frequency-inverse document frequency* (TF-IDF) sebagai extraksi fitur dan metode K Nearest Neighbor sebagai klasifikasi data teks, serta melakukan analisa hasil kinerja dan akurasi dari system klasifikasi yang dibangun serta membantu proses recruitmen pelamar kerja.

## 2. Studi Terkait

### 2.1. Klasifikasi Teks

Klasifikasi teks merupakan proses menemukan kesamaan dalam dokumen, corpus, maupun kelompok-kelompok dari dokumen yang telah dilabeli sebelumnya (supervised learning), berdasarkan topik, tema yang ditunjukkan oleh koleksi dokumen [15]. Klasifikasi teks terbagi menjadi 2 yaitu supervised dan unsupervised. Klasifikasi supervised/terbimbing merupakan proses yang dilakukan dengan arahan/bantuan expert, dimana jenis class classification ditetapkan berdasarkan class signature yang diperoleh dari training area. Sedangkan klasifikasi unsupervised/tidak terbimbing adalah proses klasifikasi yang tidak menggunakan label kelas pada data training. Sebagian besar pembentukan kelasnya dilakukan oleh computer.

### 2.2. Prepocessing

Karena ketidak strukturan data teks, maka text mining memerlukan beberapa tahan awal yang nantinya akan mempersiapkan sebuah teks agar dapat diubah menjadi lebiih terstruktur. Salah satu implementasi pada text mining adalah tahap text preprocessing. Tahap text preprocessing merupakan tahaan dimana aplikasi akan melakukan seleksi data yang nantinya diproses pada setiap dokumen. Berikut beberapa tahapan yang meliputi preprocessing:

*Case folding* : megubah semua huruf dalam dokumen teks menjadi lowercase/huruf kecil. Contoh user ingin mendapatkan informasi "SEPATU" dan mengetik "SEPATUE", "SePaTu1" atau "sepatu", tetap diberikan retrieval yang sama yaitu "sepatu" . karakter selain huruf akan dihilangkan dan dianggap delimitertriya [9].
*Tokenizing :* merupakan tahapan pemecah sekumpulan karakter dalam suatu teks ke dalam satuan kata. Sebagai contoh karakter whitespace, tabulasi, enter, dianggap sebagai pemisah kata.
*Stowpword :* atau filtering merupakan tahap mengambil kata-kata penting dari hasil token. Contoh stopword Bahasa Indonesia yaitu "yang", "dan", "dari", "di" dan lainnya.
*Stemming* :tahapan mengembalikakn kata ke bentuk dasar(root word) dengan menghilangkan aditif yang ada [9].

### 2.3. K-Nearest Neighbor

Algoritma k-Nearest Neighbor adalah algoritma supervised learning untuk memprediksi sebuah data tergolong kedalam *class* yang sama dengan *class* dari mayoritas tetangga terdekat. Untuk data yang akan diprediksi tentukan berapa k dari tetangga terdekatnya lalu melihat *class* mayoritas dari k tetangga tersebut. Tujuan dari algoritma ini adalah untuk mengklasifikasikan objek baru berdasarkan atribut dan sample-sample dari training data. Algoritma k-Nearest Neighbor menggunakan Neighborhood classification sebagain nilai prediksi dari nilai instance yang baru. Untuk menentukan jarak tetangga terdekat pada kasus ini menggunakan cosine similarity. Cosine similarity berfungi sebagai pembanding kemiripan antar dokumen. Dalam hal ini yang dibandingkan adalah query dengan dokumen latih. Setelah mendapatkan hasil dari Cosine similarity digunakan algoritma KNN untuk melakukan classifikasi Data.

### 2.4. *Voice to text/Speech to to text*

Hasil dari wawancara yang berupa audio akan dilakukan proses Transcribe(mengubah audio menjadi text). Pada proses ini digunakanya *google API* yang akan memproses hasil wawancara yang berupa audio menjadi text secara real time.

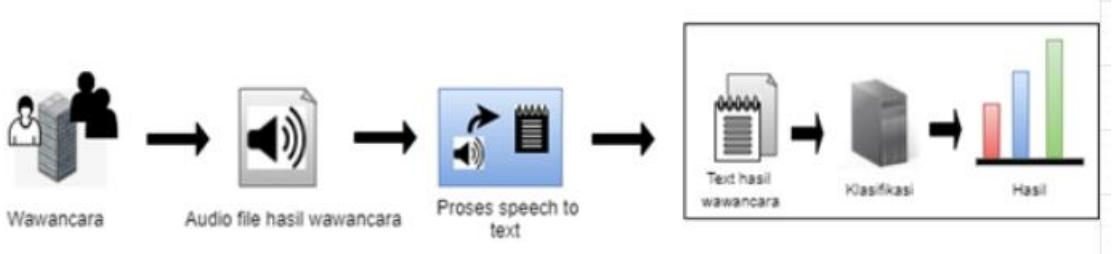
### 2.5. *Term Frequency-Inverse Document Frequency* (TF-IDF)

Setelah dilakukannya tahapan preprocessing maka akan didapatkan kumpulan term yang akan dijadikan sebagai indeks. Indeks tersebut merupakan perwakilan dari dokumen dan merupakan fitur dari dokumen tersebut. Fitur yang digunakan yaitu Unigram, dimana setiap kemunculan kata tidak bergantung kepada kata sebelumnya. Contoh "saya pergi makan" maka dari hasil kalimat tersebut menghasilkan 3 fitur yaitu 'saya', 'pergi', dan 'makan'. pengambilan Term Weighting(TF-IDF) merupakan proses pemberian nilai pada term/fitur dengan melakukan perhitungan nilai term frequency(TF), lalu melakukan perhitungan nilai *Inverse Document Frecuency* (IDF) dan melakukan perhitungan TF-IDF [6]. Nilai bobot/nilai dari setiap fitur, yang telah dihitung tersebut nantinya digunakan untuk proses selanjutnya yaitu normalisasi bobot. Nilai normalisasi bobot digunakan untuk menghitung cosine similarity pada k-nearest neighbor [6].

## 3. Sistem yang Dibangun

### 3.1. Persiapan Data

Pada proses ini sumber data yang digunakan dalam penelitian diambil dari hasil wawancara pelamar pada PT.Telkom, untuk proses yaitu pelamar mengirimkan surat lamaran dan akan dilakukan penilaian oleh divisi SDM apakah pelamar tersebut lolos ke tahap berikutnya atau tidak, jika pelamar tersebut memenuhi syarat maka pelamar akan lanjut pada tahap wawancara. Pada proses wawancara akan didapatkan audio file hasil wawancara, yang kemudian audio file tersebut akan diubah menjadi teks dengan bantuan google API *speech to text*. Setelah mendapatkan hasil dari *speech to text* yang berupa teks , kemudian data teks tersebut akan dikelompokan, diberi label dan dikonversi dari skor ke kelas nominal secara manual oleh tim ahli dengan kategorinya masing-masing.

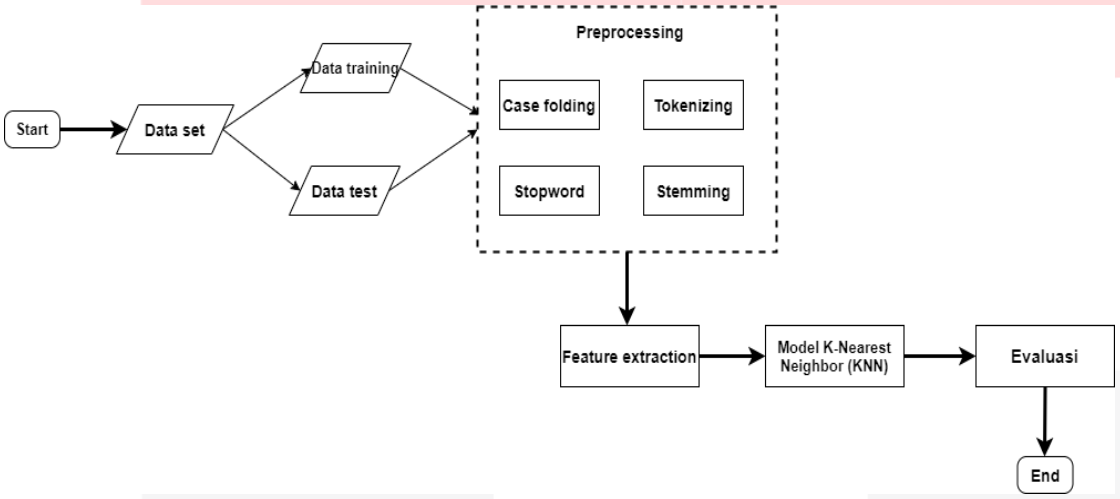



Gambar 1 Proses pengumpulan dataset

Kumpulan data teks yang merupakan hasil wawancara dibuat menjadi satu dokumen / string panjang dimana data inilah yang akan diprediski atau dinilai berdasarkan 9 (Sembilan) core value : yaitu *action, enthusiasm, focus, imagine, integrity, smart, solid, speed, dan totality*. Dan pelabelan yang dilakukan dengan memberi poin kepada setiap *core value* dengan nilai 1(tidak layak) dan 2 (layak).

Jumlah data yang diambil dari hasil wanwancara pelamar yaitu 54 data. Dari kumpulan data tersebut dilakukan cross validation dengan fold = 5,dengan perbadingannya yaitu 80:20. Hasil dari Nilai atau predeksi dari 9(sembilan) *core values* tersebut akan dijadikan sebagai acuan keputusan diterima atau tidak diterimanya pelamar kerja.

### 3.2. Struktur Model




Gambar 2 Tahapan-Tahapan Struktur Model

Dataset yang sudah dibagi menjadi data latih dan data testing akan memasuki preprocessing. Preprocessing bertujuan untuk mengubah data agar mempermudah pada saat diolah, terutama dalam menghilangkan noise, memperjelas fitur data, merubah data asli agar diperoleh data yang sesuai dengan kebutuhan. Dalam penelitian ini proses preprocessing yang dilakukan yaitu : Case folding, tokenizing, stopword(filtering), dan stemming.

### 3.3. Pelatihan Model dengan Klasifikasi

Pada penelitian ini digunakannya pelatihan *supervised learning* dengan algoritma pengklasifikasian *K-Nearest Neighbor* (KNN) dengan cosine simularity. Tahapan pelatihan model algoritma menggunakan data latih yang sudah dibagi dari dataset. Proses pengambilan data latih ke model dilakukan secara cross validation.. Pada kasus ini untuk menghitung jarak terdekat penulis menggunakan cosine similarity. Terdapat 53 data teks yang telah dilakukan labeling oleh expert yang terdiri dari 2 kelas yaitu (1) ‘layak’ dan 2 ‘tidak layak’.

Penyelesaian :

- Langkah 1
  Melakukan proprecessing terhadap 54 dokumen yang terlibat dan 1 dokumen yang akan diklasfikasikan. Melakukan *case_folding, tokenize, stopword, dan stemming*.

*Table 1 prepocesssing* data

| ***Content*** |
|---|
| Bukan tipe orang suka langgar etika orang butuh bantu… |
| Jujur pernah laku sama sikap pasti rupa buah buruk pengaruh… |
| Usaha laku jadi mahasiswa prestasi benar kan prestasi kan beda… |
| Pernah waktu muda sering masuk kuliah ikut semua atur.. |
| Pernah sih waktu sma tuh tadi masalah panitaia pernah langar.. |

- Langkah 2
  Melakukan penghitungan bobot *tf-idf* dari hasil *preprocessing.*

$$TF\text{-}IDF = tf*Log(d/df) \quad (1)$$

tf = frekuensi kemunculan term dalam dokumen
d = jumlah dokumen
df = jumlah document yang mengandung term

*Table 2* Perhitungan jumlah IDF

| Term | D1 | D2 | D3 | D4 | D5 | D6 | … | D54 | Df | d/df | idf |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Bukan | 1 | 0 | 0 | 2 | 1 | 0 | ... | 0 | 11 | 4,909 | 0,691 |
| Tipe | 1 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 2 | 27 | 1,43 |
| Orang | 4 | 1 | 0 | 1 | 0 | 0 | ... | 0 | 19 | 2,84 | 0,45 |
| Suka | 1 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 3 | 18 | 1,255 |
| Langgar | 1 | 0 | 0 | 1 | 1 | 0 | ... | 0 | 18 | 3 | 0,477 |
| Etika | 2 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 12 | 4,5 | 0,653 |
| Saat | 1 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 2 | 27 | 1,431 |
| butuh | 1 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 3 | 18 | 1,255 |

*Table 3* Perhitungan bobot TF-IDF

| Tf*idf | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | … | D54 |
| 0,691 | 0 | 0 | 1,382 | 0,691 | 0 | 0 | 0 | … | 0 |
| 1,431 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | … | 0 |
| 1,814 | 0,453 | 0 | 0,453 | 0 | 0 | 0 | 0,907 | … | 0 |
| 1,255 | 0 | 0 | 0 | 0 | 0 | 2,51 | 0 | … | 0 |
| 0,447 | 0 | 0 | 0,477 | 0,477 | 0 | 0,477 | 0 | … | 0 |
| 1,306 | 0 | 0 | 0 | 0.653 | 0 | 0,653 | 0 | … | 0 |
| 1,431 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | … | 0 |

- Langkah 3
  Menghitung Kemiripan document dengan cosine similarity

$$similarity = x*y/(\| x \| * \| y \|) \quad (2)$$

$$\|X\| = \sqrt{X_1^2 + X_2^2 + \cdots + X_n^2} \quad (3)$$

x = frekuensi dokum
y = frekuensi dokumen
|| x || = panjang vector dokumen
|| y || = panjang vector dokumen

*Table 4* Perkalian vektor

| Wd1*wdi | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | … | D54 |
| 0,447 | 0 | 0 | 0,954 | 0,447 | 0 | 0 | 0 | … | 0 |
| 2,048 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | … | 0 |
| 3,292 | 0,823 | 0 | 0,823 | 0 | 0 | 0 | 1,646 | … | 0 |
| 1,575 | 0 | 0 | 0 | 0 | 0 | 3,151 | 0 | … | 0 |
| 0,227 | 0 | 0 | 0,277 | 0,277 | 0 | 0,277 | 0 | … | 0 |
| 1,706 | 0 | 0 | 0 | 0,853 | 0 | 0,853 | 0 | … | 0 |
| 2,048 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | … | 0 |
| 109,01 | 1,075 | 12,032 | 3,691 | 5,584 | 5,214 | 10,397 | 3,78 | … | 117,613 |

*Table 5* Hitung panjang vektor

| | Panjang Vector | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | … | D54 |
| | 0,447 | 0 | 0 | 1,909 | 0,447 | 0 | 0 | 0 | … | 0 |
| | 2,048 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | … | 0 |
| | 3,292 | 0,205 | 0 | 0,205 | 0 | 0 | 0 | 0,823 | … | 0 |
| | 1,575 | 0 | 0 | 0 | 0 | 0 | 6,302 | 0 | … | 0 |
| | 0,227 | 0 | 0 | 0,277 | 0,277 | 0 | 0,277 | 0 | … | 0 |
| | 1,706 | 0 | 0 | 0 | 0,426 | 0 | 0,426 | 0 | … | 0 |
| | 2,048 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | … | 0 |
| SUM | 109,07 | 220,108 | 12,032 | 144,48 | 103,16 | 76,475 | 98,888 | 129,276 | … | 117,613 |
| SQRT | 10,443 | 14,836 | 3,468 | 12,02 | 10,156 | 8,745 | 9,944 | 11,37 | … | 10,844 |

*Table 6* Jarak tetangga dengan *cosine simililarity*

| | D1 | D2 | D3 | D4 | D5 | D6 | D7 | … | D54 |
|---|---|---|---|---|---|---|---|---|---|
| D1 | 1 | 0,007 | 0,0025 | 0,0029 | 0,05 | 0,057 | 0,1 | … | 0,003 |

Setelah mendapatkan jumlah tetangga terdekat dengan cosine similarity , lalu menentukan jumlah tetangga terdekat (jumlah K) pada penelitian ini digunakan K = 3. Setelah menentukan jumlah tetangga terdekat, maka dokumen testing dapat ditentukan klasifikasinya. K = 3,Tetangga terdekat yaitu (D1, D34) = 0,132, (D1,D18) = 0,104, dan (D1, D48) = 0,102. Maka dari dilihat class dari Dokumen D34,D18,D48 = 2 (“layak”), maka D1 termasuk Class 2(“layak”).

### 3.4. Metode validasi

Metode validasi disini melakukan evaluasi terhadap model klasifikasi dengan melihat keakuratan metode prediksi biner melalui *confussion matrix* dan tabel akurasi dan presisi untuk model dengan beberapa parameter statistik seperti *sensitivitas(recall), precision*, dan akurasi yang nilainya akan semakin bagus bila mendekati angka 1. *Sensitivitas(recall)* adalah rasio prediksi TP dibandingkan dengan keseluruhan data yang benar positif. *Precision* adalah rasio prediksi TP dibandingkan dengan keseluruhan hasil prediksi positif. akurasi adalah rasio prediksi benar (TP dan TN) dengan keseluruhan data. Parameter yang didapatkan dari hasil *confussion matrix* akan digunakan untuk mengevaluasi model yang sudah dibuat. Parameter yang digunakan adalah : true positive (TP), false positive (FP), true negative (TN), false negative (FN).

*Table 7* Model *Confussion Matrix*

| | **Class** | |
|---|---|---|
| | **Positive** | **Negative** |
| **Positive** | TN | FP |
| **Negative** | FN | TP |

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (5)$$

$$\text{Akurasi} = \frac{TP + TN}{TP + FP + FN + TN} \quad (6)$$

## 4. Hasil dan Analisis

Pada tahap ini ditampilkan hasil percobaan pada model klasifikasi dan dataset hasil wawancara pelamar kerja dengan algoritma K-nearest neighbor. Pada penelitian ini dataset berjumlah 54 *content*, 9 (sembilan) *core value* dan 2 (dua) jenis kelas yaitu layak dan tidak layak. Dataset yang dimiliki kemudian akan diambil secara acak dibagi menjadi data latih dan data uji dengan perbandingan 80% data latih dan 20% data uji. Data latih akan dijadikan sebagai model sedangkan data uji untuk menghitung akurasi model yang dibuat.

### 4.1. Pengelolaan data

Dataset yang dimiliki adalah *content* yang berisi teks hasil wawancara pelamar kerja dan dinilai berdasarkan 9 *core value* yaitu : *action, enthusiasm, focus, imagine, integrity, smart, solid, speed* dan *totality.* Label yang diberikan kepada *core value* tersebut adalah 1 (tidak layak) dan 2 (layak). Dataset seperti Tabel 2 .

*Table 8* Deskripsi dataset

| Content | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Action | enthusiasm | focus | Imagine | Integrity | Smart | Solid | Speed | Totality |
| saya bukan tipe… | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 2 |
| jujur pernah laku sama… | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 2 |
| lapor uang sesuai… | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 2 | 2 |
| pernah sih waktu sma … | 1 | 1 | 2 | 2 | 2 | 1 | 1 | 1 | 1 |

### 4.2. Analisa dan Hasil uji

Dengan menggunakan parameter yang sudah diset sebelumnya, didapatkan akurasi untuk hasil pelatihan dan pengujian pada tabel. Ditunjukan pada Tabel 5 bahwa hasil teks klasifikasi oleh algoritma K-Nearest Neighbor memberikan akurasi yang baik. Hasilnya dalam bentuk akurasi untuk setiap data latih (*training*) dan data uji (*testing*).

*Table 9* hasil akurasi *cross validation* 1

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Action | enthusiasm | focus | Imagine | Integrity | Smart | Solid | Speed | Totality |
| *testing* | 91% | 73% | 91% | 64% | 82% | 73% | 73% | 36% | 82% |
| *training* | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% |

Dari hasil percobaan yang dilakukan terhadap Model, didapatkan akurasi rata rata untuk setiap *core value* 65.2%. pada data table, nilai akurasi yang dihasilkan algoritma K-Nearest Neighbor akan memberikan sebagai model referensi yang dapat diimplementasikan dengan baik untuk penelitian lebih lanjut. Seperti dalam tabel dapat dilihat bahwa klasifikasi algoritma K-Nearest Neighbor dalam kasus seleksi pelamar kerja dengan dataset hasil wawancara pelamar kerja PT. Telkom memiliki akurasi

65,2%..

*Table 10* Hasil Akurasi rata-rata dari semua *cross validation*

| No | Algoritma | Average accuracy |
|---|---|---|
| 1 | K-Nearest Neighbor | 65.2% |

### 4.3. Hasil Validasi

Hasil klasifikasi ini yang telah diperoleh dengan menggunakan model algoritma K-Nearest Neighbor yang akan disajikan dalam bentuk *confussion matrix*. Berdasarkan hasil validasi kinerja model yang disajikan pada Tabel 10, dapat dilihat hasil dari *confussion matrix* 9 (sembilan) *core value* menggunakan KNN. Dalam kasus data pelatihan, dapat dilihat bahwa setiap model dapat secara akurat memprediksi nilai target, yang ditujukan oleh nilai *sensitivitas(recall), precision* dan akurasi yang tinggi. Dari hasil data yang diperoleh, dari 9 *core value* yang ada *core value focus* adalah yang lebih akurat dibandingkan dengan *core value* yang lain. Ini dilihat dari nilai *recall, precision*, dan akurasi yang masing masing 0.84, 0.70 dan 0.652.

*Table 11* Hasil *Cross Validation 1*

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *recall* | 1 | 0.88 | 0,9 | 0.67 | 0.78 | 0.83 | 1 | 1 | 1 |
| *precision* | 0.90 | 0.78 | 1 | 0.86 | 1 | 0.71 | 0.67 | 0.22 | 0.8 |
| *akurasi* | 0.91 | 0.73 | 0.91 | 0.64 | 0.82 | 0.73 | 0.73 | 0.36 | 0.82 |

## 5. Kesimpulan & Saran

### 5.1. Kesimpulan

Berdasarkan hasil analisa dan uji yang dilakukan, menunjukan bahwa Sistem Seleksi pelamar kerja berhasil memperoleh prediksi layak atau tidaknya pelamar dengan menggunakan teks hasil wawancara PT. Telkom. Model ini diperoleh dari hasil pelatihan dan pengujian untuk memprediksi 9 (sembilan) *core value* dalam teks hasil wawancara dengan metode K-nearest Neighbor. Dengan menggunakan model tersebut, sekarang proses wawancara dan penilaian dapat dilakukan secara otomatis dan sesuai dengan kriteria perusahaan. Model KNN ini sudah berhasil melakukan tahapan yang diperlukan dan dapat memprediksi data teks hasil wawancara pelamar kerja, meskipun dengan data yang dipakai tidak terlalu banyak. Menurut Validasi, Regresi Logistik ini memberikan hasil yang cukup bagus, seperti nilai akurasi rata rata regresi logistik pada penelitian ini yaitu 65.2%. Dari hasil training dan latih diketahui bahwa *core value focus* memiliki performa yang terbaik, dimana *recall, precision, akurasi* masing masing 0.84, 0.70 dan 0.652.

### 5.2. Saran

Perbaikan model dapat dilakukan dengan menggunakan lebih banyak data teks hasil wawancara untuk meningkatkan akurasi prediksi dari model K-Nearest Neighbor.

# Daftar Pustaka


[1] Kadhin, Ammar. (2018). "An Evaluation of Preprocessing Techniques for Text Classification", International Journal of Computer Science and Information Security

[2] T.Saito and O. Uchida.(2017). )."Automatic Labeling for News Article Classification Based on Paragraph Vector," 2017 9th Int Conf. Inf. Technol. Electr. Eng.

[3] S. Vijayarani and Ms. J. Ilamathi and Ms. Nithya. (2015). "Preprocessing techniques for Text mining-an overview", International Journal of Computer Science & Communication Networks,Vol 5(1),7-9. 20

[4] Asriyanti Indah Pratiwi and Adiwijaya.( 2018). "On the Feature Selection and Classification Based on Information Gain for Document Sentiment Analysis," Applied Computational Intelligence and Soft Computing, vol. 2018, p. 5,

[5] Mohan, V. (2015). Prepocesssing Techniques for Text Mining-an Overview. *Bharathiar University*, 10.

[6] Nurul Dyah Mentari, M. Ali Fauzan, Lailil Muflikhah. (2018). Analisis Sentimen Kurikulum 2013 pada Sosal Media Twitter Menggunakan Metode K-Nearest Neighbor dan Feature Selection Query Expansion Ranking. *Pengembangan Teknologi Informasi dan Ilmu Komputer*, 5.

[7] C. C. Aggarwal. (2012). A SURVEY OF TEXT CLASSIFICATION ALGORITHMS. Springer.

[8] J. L. A. Rajaraman. (2014).Mining of Massive Datasets,.

[9] Rut Samuel, Ripa Natan, Fitria, Ummi Syafiqoh.(2018). Penerapan Cosine Similarity dan K-Neares Neighbor(KNN) pada Klasifikasi dan Pencarian Buku. STMIK PPKIA Tarakanita Rahmawati.

[10] Celli, F., Pianesi, F., Stillwell, D. S., and Kosinski, M. (2013). Workshop on Computational Personality Re-cognition (Shared Task). The Seventh International AAAI Conference on Weblogsand Social Media. Boston, MA, USA.

[11] Rout, D.; Preot iuc-Pietro, D.; Kalina, B.; and Cohn, T.( 2013). Where's @wally: A Classification Approach to Geolocating Users based on their Social Ties. HT, 11–20.

[12] APJII, "Jumlah pengguna internet 2017 meningkat, kominfo terus lakukan percepatan pembangunan broa- dband," 2017.

[13] Dewi, Sari. (209). "Komparisasi 5 Metode Algoritma Klasifikasi Data Mining pada Prediksi Keberhasilan Pemasaran Produk Layanan Perbankan", Jurnal Techno Nusa Mandiri Vol. XIII, No.1.

[14] C.D. Manning, P. Raghavan, H. Schutze. (2008). Introduction to Information Retrieval. Cambridge UP

[15] T. Hastie, R. Tibshirani, and J. Friedman. (2009). The Elements of Statistical Learning. Springer Verlag, 2 edition.

[16] Chakraborty et al. 2013. Text Mining and Analysis, Practical Methods,Examples and Case Studies Using SAS. North Carolina : SAS Instititute Inc.

# Lampiran

Tabel 1 *Cross Validation* 1

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *testing* | 91% | 73% | 91% | 64% | 82% | 73% | 73% | 36% | 82% |
| *training* | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% |

Tabel 2 *Cross validation* 2

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *testing* | 82% | 54% | 82% | 64% | 36% | 64% | 82% | 45% | 72% |
| *training* | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% |

Tabel 3 *Cross validation* 3

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *testing* | 82% | 73% | 64% | 64% | 73% | 55% | 45% | 45% | 64% |
| *training* | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% |

Tabel 4 *Cross validation* 4

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *testing* | 73% | 45% | 82% | 55% | 64% | 45% | 45% | 45% | 100% |
| *training* | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% |

Tabel *5 Cross validation* 5

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *testing* | 90% | 70% | 80% | 60% | 50% | 50% | 50% | 30% | 90% |
| *training* | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% |

Tabel 6 *Cross validation recall* dan *precision* 1

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *recall* | 1 | 0.88 | 0,9 | 0.67 | 0.78 | 0.83 | 1 | 1 | 1 |
| *precision* | 0.90 | 0.78 | 1 | 0.86 | 1 | 0.71 | 0.67 | 0.22 | 0.8 |
| *akurasi* | 0.91 | 0.73 | 0.91 | 0.64 | 0.82 | 0.73 | 0.73 | 0.36 | 0.82 |

Tabel 7 *Cross validation recall* dan *precision* 2

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *recall* | 0.90 | 0.71 | 1 | 0.67 | 0.5 | 0.7 | 1 | 0.5 | 1 |
| *precision* | 0.90 | 0.62 | 0,82 | 0.86 | 0.43 | 0.88 | 0.82 | 0.83 | 0.73 |
| akurasi | 0.82 | 0.54 | 0.82 | 0.64 | 0.36 | 0.64 | 0.82 | 0.45 | 0.72 |

Tabel 8 *Cross validation recall* dan *precision* 3

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *recall* | 0.82 | 1 | 0,88 | 0.7 | 1 | 0.83 | 0,71 | 0,43 | 1 |
| *precision* | 1 | 0.62 | 0,7 | 0.88 | 0,62 | 0.56 | 0.56 | 0.6 | 0.64 |
| *akurasi* | 0.82 | 0.73 | 0.64 | 0.64 | 0.73 | 0.55 | 0.45 | 0.45 | 0.64 |

Tabel 9 *Cross validation recall* dan *precision* 4

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *recall* | 1 | 0.83 | 1 | 0.71 | 1 | 0.57 | 0.62 | 0.62 | 1 |
| *precision* | 0.73 | 0.5 | 0.82 | 0.62 | 0.56 | 0.57 | 0.62 | 0.62 | 1 |
| *akurasi* | 0.73 | 0.45 | 0.82 | 0.55 | 0.64 | 0.45 | 0.45 | 0.45 | 100 |

Tabel 10 *Cross validation recall* dan *precision* 5

| | Core Values | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Action** | **enthusiasm** | **focus** | **Imagine** | **Integrity** | **Smart** | **Solid** | **Speed** | **Totality** |
| *recall* | 1 | 0.88 | 0,8 | 1 | 1 | 0.8 | 0.8 | 1 | 1 |
| *precision* | 0.90 | 0.78 | 1 | 0.56 | 0.44 | 0.5 | 0.5 | 0.22 | 0.9 |
| *akurasi* | 0.90 | 0.70 | 0.80 | 0.60 | 0.50 | 0.50 | 0.50 | 0.30 | 0.90 |