IMPLEMENTASI K NEAREST NEIGHBOR (KNN) PADA KLASIFIKASI ARTIKEL WIKIPEDIA INDONESIA

Erik Hardiyanto[1], Faisal Rahutomo[2], Dwi Puspitasari[3]

Jurusan Teknologi Informasi, Program Studi Teknik Informatika, Politeknik Negeri Malang
Email: [1] hardiyantoerik@gmail.com, [2] faisal.polinema@gmail.com, [3] dwi_sti@yahoo.com

Abstrak

Suatu hal yang dibutuhkan seiring dengan perkembangan teknologi informasi dan komunikasi adalah informasi. Salah satu sumber informasi tersebut adalah Wikipedia Bahasa Indonesia. Banyaknya artikel yang masuk dalam beberapa kategori menyebabkan pembaca kesulitan dalam mencari informasi, terutama dalam pencarian berdasarkan kategori. Oleh karena itu diperlukan sebuah klasifikasi untuk artikel Wikipedia agar memiliki tepat satu kategori namun tetap dapat berhubungan dengan kategori lainnya. Diperlukan sistem yang dapat mengklasifikasi artikel Wikipedia Indonesia secara otomatis. Klasifikasi artikel Wikipedia Indonesia adalah sebuah sistem yang berfungsi untuk mengklasifikasi artikel Wikipedia Indonesia yang berupa dokumen teks dengan tahapan *text preprocessing* dilanjutkan dengan pembobotan *TF IDF* pada masing-masing artikel Wikipedia Indonesia terbentuk vektor kata. Berdasarkan pembobotan tersebut, artikel-artikel Wikipedia Indonesia tersebut diklasifikasikan dengan metode K Nearest Neighbor. Perhitungan centroid pada masing-maing sub sub kategori terdiri dari tiga buah artikel yang diambil nilai tengahnya kemudian dihitung jarak kedekatan dengan masing-masing data uji. Berdasarkan hasil pengujian manual menunjukkan akurasi kebenaran sebesar 60%.

Kata kunci*: text preprocessing, pembobotan TF IDF,* vektor kata*, K Nearest Neighbor.*

1. *Pendahuluan*

Wikipedia merupakan ensiklopedia elektronik terbesar di dunia saat ini(Wang, 2008). Wikipedia Indonesia adalah versi Bahasa Indonesia dari ensiklopedia. Wikipedia sebagai ensiklopedia yang dapat disunting bebas oleh siapa saja melalui jaringan Internet. Wikipedia Indonesia memiliki 371.150 lebih artikel (sumber: https://id.wikipedia.org/wiki/Wikipedia_bahasa_Indonesia). Satu artikel Wikipedia Indonesia bukan hanya untuk satu kategori, melainkan beberapa kategori. Sebagai contoh artikel bahasa Jawa dapat termasuk dalam kategori bahasa, Bahasa Indonesia, dan bahasa daerah.

Hal tersebut dapat menyebabkan pembaca kesulitan dalam mencari informasi, terutama dalam pencarian berdasarkan kategori. Oleh karena itu diperlukan sebuah klasifikasi untuk artikel Wikipedia agar memiliki tepat satu kategori namun tetap dapat berhubungan dengan kategori lainnya. Dalam permasalahan seperti ini, penulis menggunakan. Metode tersebut adalah *K Nearest Neighbor* (*KNN*). Metode *KNN* merupakan metode untuk data yang sebelumnya telah memiliki kelas, oleh karena itu diperlukan data latih dan data uji. Metode *KNN* dapat melakukan klasifikasi terhadap dokumen-dokumen yang telah menghasilkan nilai similaritas(Purwanti, 2015). Perhitungan similaritas tersebut menggunakan pendekatan euclidean distance.

Banyaknya jumlah artikel Wikipedia yang akan diklasifikasi membutuhkan proses klasifikasi dengan waktu yang lama. Oleh karena itu diperlukan sistem yang dapat mengklasifikasi artikel Wikipedia secara otomatis dengan menggunakan metode *text preprocessing* untuk mengolah teks pada artikel Wikipedia sehingga hasil pengolahan teks tersebut dapat dimanfaatkan untuk klasfikasi artikel menggunakan metode *KNN*.

2. *Text Preprocessing*

Dikarenakan dokumen teks memiliki data yang tidak terstruktur maka digunakanlah *text preprocessing* ini untuk merubah data yang belum terstruktur itu menjadi sebuah data yang terstruktur sehingga dapat siap untuk digunakan dalam proses selanjutnya. *Text Preprocessing* ini memiliki beberapa tahapan yaitu (Nugroho, 2016):

a. Mengekstrak teks yang akan kita olah.
b. Melakukan stopword, yaitu menghilangkan kata-kata yang tidak bermakna misalkan kata hubung.

3. *Pembobotan* TF IDF

Term Frequency dan *Inverse Document Frequency* (*TF IDF*) merupakan pembobotan yang sering digunakan dalam penelusuran informasi dan *text mining* (Turney dkk, 2010). Term frequency adalah pembobotan yang sederhana dimana penting tidaknya sebuah kata dianggap sama atau sebanding dengan jumlah kemunculan kata tersebut dalam dokumen, sementara itu *inverse document frequency* (*IDF*) adalah pembobotan yang mengukur penting

sebuah kata dalam dokumen dilihat pada seluruh dokumen secara global (Purwanti, 2015)rumus:

$$W_{(t,f)} = TF_{(t,d)} \times IDF_{(t)} \quad (1)$$

Dimana nilai$IDF_{(t)}$didapat dari:

$$IDF_{(t)} = \log(\frac{|D|}{df_{(t)}}) \quad (2)$$

Keterangan:

$TF_{(t,d)}$: Jumlah kemunculan token $t$pada dokumen$d$

$IDF_{(t)}$: Nilai *IDF* token t

$df_{(t)}$: Jumlah dokumen yang memuat token t

$|D|$: Jumlah dokumen dalam korpus

4. *K Nearest Neighbor*

Algoritma *K Nearest Neighbor* merupakan sebuah algoritma yang sering digunakan untuk klasifikasi teks dan data(Samuel, 2014). Metode *KNN* adalah metode yang melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan obyek tersebut. *KNN* termasuk algoritma supervised learning(Nugroho, 2016). *Supervised learning* merupakan suatu pembelajaran yang terawasi dimana jika output yang diharapkan telah diketahui sebelumnya. Biasanya pembelajaran ini dilakukan dengan menggunakan data yang telah ada. Pada metode ini, setiap pola yang diberikan ke dalam telah diketahui outputnya.

Tujuan dari algoritma ini adalah mengklasifikasikan obyek berdasarkan atribut dan sampel data latih. Apabila algoritma tersebut diberikan titik query, maka akan ditemukan sejumlah k obyek atau titik latih yang paling dekat dengan titik query. Klasifikasi menggunakan voting terbanyak diantara klasifikasi dari k obyek. Algoritma *KNN* menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari *query instance* yang baru.

Algoritma *KNN* sangat sederhana, bekerja berdasarkan jarak terpendek dari query instance ke sampel data latih untuk menentukan *KNN*. Sampel data latih diproyeksikan ke ruang berdimensi banyak, dimana masing-masing dimensi merepresentasikan fitur dari data. Ruang ini dibagi menjadi bagian-bagian berdasarkan klasifikasi training sample. Sebuah titik pada ruang ini ditandai jika merupakan klasifikasi yang paling banyak ditemukan pada k buah tetangga terdekat dari titik tersebut. Metode pencarian jarak yang digunakan adalah Euclidean Distance yaitu perhitungan jarak terdekat. Perhitungan jarak terdekat dibutuhkan untuk menentukan jumlah kemiripan yang dihitung dari kemiripan kemunculan teks yang dimiliki suatu paragraf. Setelah itu kemunculan teks yang sedang diujikan dibandingkan terhadap masing-masing sampel data asli.

Persamaan jarak *euclidean distance*:

$$d_{(i,j)} = \sqrt{(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \cdots + |x_{ip} - x_{jp}|^2)} \quad (3)$$

Dengan:

$d_{(i,j)}$: Jarak dokumen ke-i ke dokumen ke-j

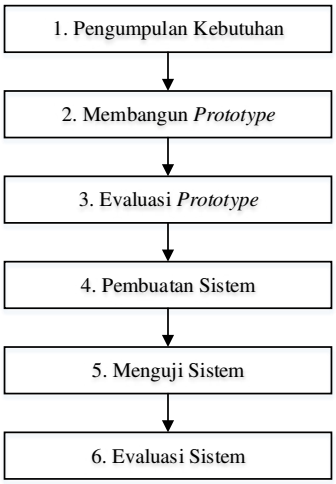
$x_{i(n)}$: Kata ke n di dokumen ke-i

$x_{j(n)}$: Kata ke n di dokumen ke-j

5. Metodologi Penelitian

Pada bagian ini dibahas metode yang digunakan peneliti dalam Implementasi KNN pada klasifikasi artikel Wikipedia Indonesia. Metode pengembangan aplikasi yang digunakan adalah metode prototyping.

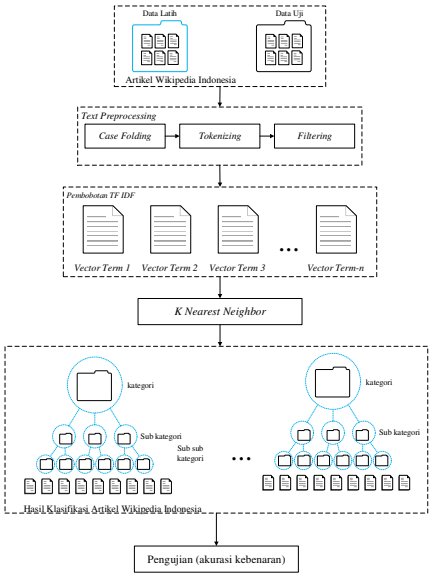
Proses kegiatan yang ada pada metode prototyping dapat dijelaskan pada Gambar 1.



Gambar 1.*Prototype Model*

6. Rancangan

Klasifikasi artikel Wikipedia Indonesia adalah sebuah sistem yang berfungsi untuk mengklasifikasi artikel Wikipedia Indonesia yang berupa dokumen teks. Tahapan *text preprocessing* dilanjutkan dengan pembobotan *TF IDF* pada masing-masing artikel Wikipedia Indonesia hingga terbentuk vektor kata. Berdasarkan pembobotan tersebut, artikel-artikel Wikipedia Indonesia tersebut diklasifikasikan dengan metode *KNN*. Hasil klasifikasi tersebut mengelompokan suatu artikel masuk ke sebuah kelas tertentu. Hasilnya, artikel yang memiliki nilai jarak terkecil dengan kelas yang sudah ditentukan dimasukan pada kelas tersebut. Rancangan sistem yang dibangun ditujukan pada gambar 2.

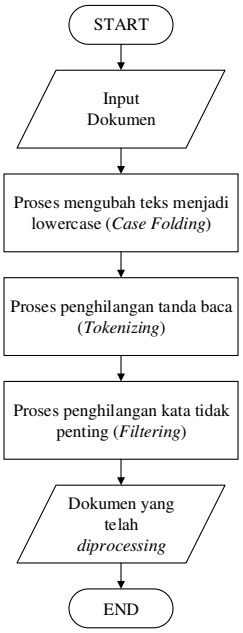


Gambar 2. Rancangan Sistem

7. Diagram Alir

7.1 Diagram Alir *Text Preprocessing*

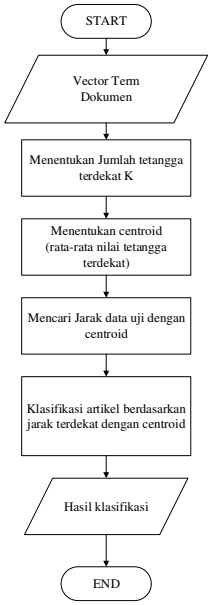
Diagram alir tahap preprocessing merupakan diagram alir yang berisi proses penghilangan tanda baca (tokenization) serta proses penghilangan kata yang tidak penting (stopwords). Proses ini dijelaskan lebih lanjut pada gambar 3.



Gambar 3. Diagram Alir *Text Preprocessing*

7.2 Diagram Alir KNN

Diagram alir Algoritma *KNN*merupakan diagram alir yang berisi urutan masukan berupa vektor kata dari masing-masing dokumen yaitu bobot yang dimiliki setiap kata/term, menentukan jumlah tetangga terdekat, menentukan *centroid* (rata-rata nilai tetangga terdekat), mencari jarak data uji dengan *centroid*, klasifikasi artikel berdasarkan jarak terdekat dengan *centroid*.

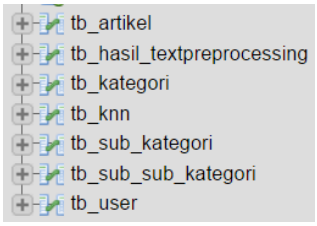


Gambar 4.Diagram Alir *KNN*

8. Implementasi

8.1 Implementasi Basis Data

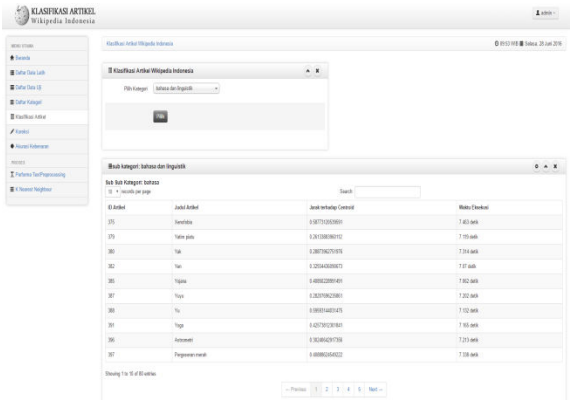
Berikut ini merupakan hasil implementasi basis data:



Gambar 5. Implementasi Basis Data

8.2 Implementasi Antarmuka

Berikut ini merupakan hasil implementasi antarmuka sistem:



Gambar 6. Implementasi Antarmuka

9. Uji Coba

Untuk menguji metode *KNN* pada klasifikasi artikel Wikipedia dibutuhkan responden untuk mengoreksi kebenaran suatu artikel Wikipedia. Kebenaran yang dimaksud adalah kebenaran hasil klasifikasi, apakah sudah sesuai pada kategori masing-masing artikel. Langkah pertama dari pengujian ini adalah memilih sub kategori dari masing-masing kategori artikel, kemudian sub kategori akan menampilkan sub sub kategori serta artikel yang ada di dalamnya.

Langkah berikutnya adalah koreksi terhadap artikel Wikipedia dalam satu sub kategori. Koreksi tersebut terdapat dua pilihan yaitu benar atau salah. Kemudian artikel yang benar dijumlahkan sehingga mendapatkan jumlah artikel yang benar untuk selanjutnya dihitung menjadi prosentase akurasi kebenaran artikel.

10. Pengujian *Text Preprocessing*

Pada tahap *text preprocessing*, artikel Wikipedia menjalani proses *case folding*, *tokenizing*, dan *filtering*. Setelah proses tersebut dijalankan, kemudian setiap kata/term dilakukan pembobotan *TF IDF* sehingga terbentuk vektor kata. Kedua tahap tersebut dieksekusi bersama pada data artikel. Berikut hasil eksekusi dari masing-masing data artikel:

Artikel	Waktu Eksekusi *TF*	Waktu Eksekusi *TF IDF*
1-100	35.29 detik	39.916 detik
101-200	36.563 detik	37.519 detik
201-300	42.431 detik	35.739 detik
301-400	35.53 detik	35.689 detik
401-500	35.618 detik	36.226 detik
501-572	29.761 detik	26.852 detik

Tabel 1. Pengujian *Text Preprocessing*

Sumber: Pengujian *Text Preprocessing*

11. Pengujian*KNN*

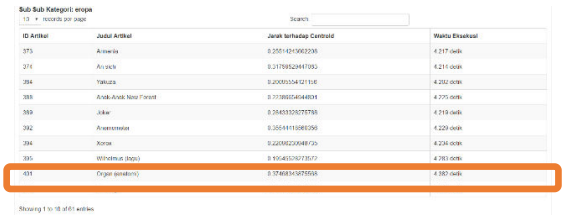
Berdasarkan pengujian sistem yang telah dilakukan, dapat diketahui bahwa sistem pada penelitian ini telah berjalan dengan baik secara fungsional dan menghasilkan output yang diharapkan. Berdasarkan hasil pengujian masing-masing sub kategori, maka dilakukan perhitungan untuk mencari persentase akurasi kebenaran. Berikut ini adalah perhitungan mencari akurasi kebenaran.

$$\text{Akurasi Kebenaran (\%)} = \frac{\sum Koreksi\ Benar}{\sum Data\ Uji}$$

$$= \frac{80}{200} = 60\%$$

Terdapat faktor-faktor yang mempengaruhi akurasi kebenaran tersebut yaitu data uji diambil secara acak dari file dump Wikipedia, sehingga belum tentu semua data uji dapat masuk ke dalam kategori yang sesuai. Selain itu isi dari masing-masing data uji juga berpengaruh terhadap klasifikasi. Jika judul dari artikel sesuai, namun isinya tidak sesuai maka tidak akan masuk ke dalam sub sub kategori yang sesuai dengan judul artikel. Faktor berikutnya adalah jumlah artikel yang akan dijadikan *centroid*, jika semakin banyak jumlah artikel maka *centroid* akan semakin akurat.

Sebagai contoh artikel yang berjudul "organ" pada gambar 5 masuk ke dalam sub sub kategori eropa, sub kategori geografi dan tempat-tempat,dan kategori geografi. Artikel ini seharusnya masuk kedalam kategori ilmu alam. Namun karena artikel "organ" memiliki jarak paling dekat dengan *centroid* sub sub kategori eropa sehingga artikel "organ" masuk kedalam sub sub kategori eropa.



Gambar 7. Contoh Pembahasan Pegujian

Untuk Sub sub kategori Eropa, artikel-artikel yang membentuk centroidnya adalah artikel yang berjudul "Belanda", "Belgia", "Jerman".

12. Kesimpulandan Saran

Berdasarkan pembahasan dapat ditarik beberapa kesimpulan: (1) Semua data uji telah masuk ke dalam setiap sub sub kategori pada masing-masing sub kategori dan kategori namun tidak semua artikel sesuai dengan sub sub kategori, oleh karena itu diperlukan pengujian manual. (2) *Centroid* pada masing-maing sub sub kategori terdiri dari tiga buah artikel yang diambil nilai tengahnya kemudian dihitung jarak kedekatan dengan masing-masing data uji. (3) Berdasarkan hasil pengujian manual menunjukkan akurasi kebenaran sebesar 60%. Terdapat faktor-faktor yang mempengaruhi akurasi kebenaran tersebut yaitu data uji diambil secara acak dari file dump Wikipedia, sehingga belum tentu semua data uji dapat masuk ke dalam kategori yang sesuai. Selain itu isi dari masing-masing data uji juga berpengaruh terhadap klasifikasi. Jika judul dari artikel sesuai, namun isinya tidak sesuai maka tidak akan masuk ke dalam sub sub kategori yang sesuai dengan judul artikel. Faktor berikutnya adalah jumlah artikel yang akan dijadikan *centroid*, jika semakin banyak jumlah artikel maka *centroid* akan semakin akurat.

Daftar Pustaka:

Nugroho, Moh Aziz dan Santoso, Heru Agus, 2016. "Klasifikasi Dokumen Komentar Pada Situs

Youtube Menggunakan Algoritma K-Nearest Neighbor (K-NN)". Jurnal Sistem Informasi

Purwanti, Endah., 2015. "Klasifikasi Dokumen Temu Kembali Informasi dengan K-Nearest Neghbour".e-ISSN 2442-5168. 1(2), 129-138

Samuel, Yoseph. Dkk, 2014. "Implementasi Metode K-Nearest Neighbor dengan Decision Rule untuk Klasifikasi Subtopik Berita". Jurnal Informatika. 10(1), 1-15

Turney, P. D. Pantel, dan Patrick. (2010). “From Frequency to Meaning: Vector Space Models of Semantics”. Journal of Artificial Intelegence Reseach, 37, 141188.

Wang, Pu dan Carlotta Domeniconi, 2008. “Building Semantic Kernels for Text Classification using Wikipedia”. KDD ‘08 Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. 713-721.