**IMPLEMENTASI K-NN DAN NAIVE BAYES & APLIKASINYA UNTUK KLASIFIKASI**

**Tugas Besar 2**

**IF3170 Inteligensi Buatan**

oleh: **Kelompok 5 - K2**

**Irene Wiliudarsan 13513002**

**William Sentosa 13513026**

**Angela Lynn 13513032**

**Devina Ekawati 13513088**



**TEKNIK INFORMATIKA**

**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA**

**INSTITUT TEKNOLOGI BANDUNG**

**BANDUNG**

**2015**

**1. Deskripsi Masalah**

Program yang akan dibuat harus dapat mengimplementasi algoritma k-Nearest Neighbour dan Naive Bayes untuk skema *full training* dan *10-fold cross validation*. Setelah itu, hasil program akan dibandingkan dengan *tools* weka untuk mengetahui apakah program sudah berjalan dengan benar atau tidak . Program juga harus dapat menampilkan model pembelajaran Naive bayes.

Aplikasi yang dibangun harus dapat menerima input berupa atribut sesuai dataset tertentu dan menghasilkan kelas berdasarkan model klasifikasi dari setiap teknik pembelajaran mesin. Setelah itu, aplikasi harus dibungkus dengan antarmuka yang baik agar mudah digunakan oleh pengguna.

**2. Implementasi Algoritma pada Program**

**Algoritma K-Nearest Neighbour**

Berikut proses-proses untuk pengujian satu datum dengan acuan suatu dataset, menggunakan algoritma K-Nearest Neighbour, yang diterapkan pada program ini:

1. Menerima masukan berupa data yang akan diuji.
2. Membandingkan data masukan tersebut dengan *dataset* acuan untuk mendapatkan jarak antara data masukan dengan setiap data pada *dataset*.
3. Memilih k data dengan jarak terkecil. Proses iterasi mencari k data terkecil pada *dataset*, dimulai dari atas ke bawah. Jika terdapat beberapa data dengan jarak terkecil yang sama, lebih banyak dari k, maka akan diambil data paling atas.
4. Dari k data yang sudah terpilih, dihitung jumlah frekuensi kemunculan setiap nilai kelas. Nilai kelas dengan frekuensi paling banyak akan dipilih sebagai kelas data masukan. Apabila terdapat lebih dari satu kelas yang memiliki frekuensi terbanyak, maka kelas data masukan tidak bisa diklasifikasikan.

Berikut proses-proses implementasi K-NN untuk skema *full-training*:

1. Menerima masukan berupa suatu *dataset*.
2. Untuk setiap data pada *dataset*, dilakukan proses pengklasifikasian dengan acuan *dataset* tersebut, menggunakan algoritma K-NN. Kemudian hasil klasifikasi tersebut dibandingkan dengan kelas data pada *dataset*.
3. Semakin banyak hasil klasifikasi yang cocok dengan kelas data pada *dataset*, maka semakin tinggi akurasinya.

Berikut proses-proses implementasi K-NN untuk skema *ten-fold cross validation*:

1. Menerima masukan berupa *dataset*.
2. *Dataset* dibagi menjadi 10 *fold*. Untuk setiap data pada satu *fold*, dilakukan proses pengklasifikasian dengan acuan *dataset* dari 9 *fold* lainnya, menggunakan algoritma K-NN. Kemudian hasil klasifikasi tersebut dibandingkan dengan kelas data pada *dataset*.
3. Semakin banyak hasil klasifikasi yang cocok dengan kelas data pada *dataset*, maka semakin tinggi akurasinya.

**Algoritma Naïve Bayes**

Berikut merupakan langkah-langkah yang dilakukan untuk mendapatkan model pembelajaran *Naïve Bayes*.

1. **Menghitung jumlah frekuensi setiap atributnya berdasarkan kelas**

Sebagai contoh pada data car.arff, atribut *buying* dapat berisi dengan nilai *vhigh, high, med* dan *low*. Untuk setiap nilai ini, dilakukan perhitungan frekuensi berdasarkan kelas yang ada. Hasilnya disimpan dalam sebuah tabel.

1. **Membagi dengan jumlah frekuensi kelasnya untuk menghitung probabilitas**

Tabel frekuensi akan dibagi dengan frekuensi masing-masing kelas. Contoh : atribut *buying* dengan nilai *vhigh* yang tergolong dalam kelas *unacc* memiliki frekuensi 40 sedangkan kelas *unacc* secara keseluruhan memiliki frekuensi 160 sehingga probabiitas untuk nilai *vhigh* pada kelas *unacc* sebesar 0.25.

Untuk mendapatkan prediksi kelas dari atribut tertentu, perhitungan probabilitas harus dilakukan.

Cara perhitungannya:

Perhitungan perlu dilakukan untuk masing-masing kelas yang mungkin dan kelas dengan probabilitas paling besar merupakan hasilnya.

Berikut merupakan proses-proses implementasi Naive Bayes untuk pengujian full training :

1. Membuat model pembelajaran dengan menggunakan seluruh dataset yang ada.
2. Mengklasifikasikan setiap dataset yang ada dengan menggunakan model yang sudah dibangun.
3. Menghitung jumlah dataset yang klasifikasinya sesuai dan tidak sesuai dengan kelas aslinya.
4. Menghitung jumlah akurasi dengan menghitung presentase dataset yang terklasifikasi secara benar dan tidak.

Berikut merupakan proses-proses implementasi Naive Bayes untuk pengujian full training :

1. Memisahkan sepuluh persen data dengan sembilan puluh persen data lainnya.
2. Membuat model pembelajaran berdasarkan sembilan puluh persen data tersebut.
3. Mengklasifikasikan setiap dataset yang tercakup dalam sepuluh persen data tersebut dengan menggunakan model yang sudah dibangun.
4. Menghitung jumlah dataset yang klasifikasinya sesuai dan tidak sesuai dengan kelas aslinya.
5. Mengulangi langkah pertama dengan memilih sepuluh persen data lainnya dan membangun model kembali sampai seluruh dataset sudah terklasifikasi.
6. Menghitung jumlah akurasi dengan menghitung presentase dataset yang terklasifikasi secara benar dan tidak.