

KLASIFIKASI DATA TIME SERIES ARUS LALU LINTAS JANGKA PENDEK MENGGUNAKAN ALGORITMA ADABOOST DENGAN RANDOM FOREST

Ahmad Rofiqul Muslikh¹, Heru Agus Santoso², Aris Marjuni³

¹²³Pascasarjana Teknik Informatika Universitas Dian Nuswantoro

ABSTRACT

Data traffic in Indonesia is used for management control traffic flow, while the data on get results from the survey will be undertaken directly localized, the survey will be undertaken are less effective, and the data obtained from the survey results were used as a reference in control traffic flow, and therefore to obtain the data traffic flow more effective in need of a new approach that can classified and predict the data in the can with higher accuracy, so that density and congestion can be predicted earlier. In this study used the approach of using Adaboost and Random Forest algorithms to classification and predict the survey data that are time series, the results of testing for prediction using Adaboost with Random Forest With Confusion Matrix as a measuring accuracy rate of 87,8%, and the rate of error in getting at 0,0629. On the results using Adaboost with a Random Forest approach proved to be more efficient in predicting the survey data rather than simply relying on the original data to predict traffic flow.

Keywords : Traffic flow , Classification, Prediction , Adaboost, Random Forest

1. PENDAHULUAN

Manajemen lalu lintas di jalan raya sangat penting untuk transportasi. Dengan adanya kontrol manajemen lalu lintas dapat dijadikan bahan untuk mengklasifikasikan dan memprediksi arus lalu lintas, kemacetan dan kepadatan. Data yang digunakan dalam manajemen lalu lintas di Indonesia didapatkan dari survei yang dilakukan dinas Pekerjaan Umum, untuk mendapatkan data tersebut selama ini masih menggunakan cara manual yaitu dinas PU langsung mengadakan survei di tempat lokasi, maka dari itu untuk mempermudah survei arus lalu lintas jangka pendek dibutuhkan pendekatan baru agar lebih efisien dalam memprediksi arus lalu lintas.

Transportasi merupakan sektor yang penting dalam kehidupan masyarakat, karena dengan adanya transportasi membantu masyarakat untuk melakukan kegiatan ekonomi. Kegiatan ekonomi tidak akan berjalan jika tidak diseimbangkan dengan kebutuhan transportasi yang memadai. Pada penelitian [1] bahwa trend perkembangan transportasi pada saat ini masih belum memadai, atau dalam istilah lain masih belum seimbang antar pertumbuhan penduduk dengan pembangunan infrastruktur transportasi.

Negara Indonesia sebagai Negara dengan populasi yang besar masih tergolong sebagai Negara berkembang yang di dalamnya perlu adanya pembangunan yang signifikan dengan pertumbuhan penduduk di dalamnya. Masyarakat lebih memilih untuk menggunakan kendaraan pribadi daripada menggunakan angkutan umum. Pertumbuhan transportasi umum menurut BPS [2] pada tahun 2014 hanya tumbuh sebesar 0,31% berbanding terbalik dengan pertumbuhan kendaraan pribadi seperti roda dua yang tumbuh sebesar 11,08 % [2]. Agar transportasi umum menjadi pilihan utama dalam menggerakkan ekonomi Negara harus diseimbangkan dengan fasilitas umum juga, pemerintah harus lebih mengedepankan transportasi umum. Tetapi disamping itu bukan hanya pihak pemerintah saja yang harus bertanggung jawab dalam memfasilitasi transportasi umum, masyarakat juga harus mendorong program – program pemerintah dalam memberikan layanan transportasi umum tidak selalu pemerintah yang berkewajiban.

Lebih membudayanya masyarakat dalam menggunakan transportasi pribadi, seperti mobil dan motor tetapi tidak diseimbangkan dengan pertumbuhan fasilitas jalan yang pada akhirnya menjadi dampak

negatif, dampak negatif itu berupa kemacetan di jalan. Di kota besar maupun kota sedang seperti Malang pun terjadi kemacetan. Karena pertumbuhan transportasi dan pertumbuhan penduduk tidak seimbang dengan fasilitas jalan yang digunakan sekarang. Pertumbuhan penduduk mulai tahun 2010 – 2014 di Jawa Timur saja mengalami pertumbuhan sebesar 0,69% [3], meskipun pertumbuhan penduduk sebesar 0,69% di Jawa Timur tetapi pertumbuhan kendaraan bermotor seperti mobil penumpang tumbuh sekitar 9,11%, Bus 1,61%, Mobil barang 7,39% dan sepeda motor 11,08%. Masyarakat lebih memilih transportasi darat seperti mobil dan motor karena lebih efisien terhadap waktu yang ditempuh. Terkadang dalam satu keluarga saja bisa mempunyai 4 motor dan 1 mobil pribadi, bayangkan saja jika dikalikan dengan pertumbuhan jalan yang masih lebih kecil perbandingannya, akan terjadi kemacetan terjadi.

Contohnya saja di Kota Malang yang notabene sebagai kota pendidikan dan kota pariwisata, pertumbuhan jalan dan pertumbuhan transportasi masyarakat yang digunakan tidak seimbang, masyarakat lebih menggunakan kendaraan pribadi dalam melakukan aktifitasnya daripada menggunakan transportasi umum. Seperti yang saya jelaskan tadi karena lebih efisien dalam segi waktu.

Kota Malang merupakan kota yang strategis dalam pergerakan ekonomi. Secara geografis kota Malang berada di selatan dari Kota Surabaya, di sebelah barat kota Malang terdapat kota Batu. Dan kota Malang merupakan destinasi wisata. Sebagai kota destinasi wisata banyak sekali wisatawan dari dalam maupun luar kota ketika musim liburan atau musim wisata melakukan wisata ke kota Malang. Kota Batu juga merupakan pusat dari kota wisata yang ada di Jawa Timur. Letak kota Batu yang berada di sebelah barat kota Malang menjadi letak yang strategis dalam segi wisata. Dampak positifnya dapat menarik investor dalam segi wisata untuk berinvestasi dalam wisata di kota Batu maupun Malang. Disamping itu ada dampak negatif wisata yaitu tidak seimbangnya pertumbuhan wisata dengan jalan yang digunakan untuk menuju lokasi wisata.

Malang sebagai gerbang wisata mempunyai jalan yang masih belum memadai dengan tingkat kunjungan wisatawan, sebagai contoh saja wisatawan yang ingin ke kota Batu dan kota Malang mempunyai 1 jalan utama yaitu jalan Malang - Pasuruan. Dikarenakan hanya mempunyai 1 pintu masuk dari sebelah utara dan dengan laju transportasi yang ingin berwisata ke kota Malang dan kota Batu, akhirnya kapasitas jalan yang ada tidak memadai. Kota Malang yang mempunyai GDP tahun 2014 sebesar 46.563 [3] merupakan kota dengan pertumbuhan ekonomi dan pariwisata yang berkembang tiap tahunnya. Tidak memadai kapasitas jalan tersebut dikarenakan beberapa faktor diantaranya wisatawan dari utara kota Malang menuju kota Malang maupun kota Batu menggunakan 1 jalan saja, terlebih lagi faktor penduduk Malang yang juga melakukan aktifitas di jalan raya, akhirnya 1 jalan tersebut tidak bisa menampung jumlah kendaraan yang ada dan menjadikan gerbang masuk kota Malang menjadi tempat kemacetan yang parah. Terlebih lagi jika pada musim liburan, meningkatnya wisatawan yang ingin berwisata ke Malang dan kota Batu bertemu dengan arus dari penduduk kota Malang sendiri kemacetan lebih parah lagi.

Fungsi Jalan pada kota Malang dibagi menjadi 3 jalan, Arteri Primer, Arteri Sekunder, Lokal Primer, dan Lokal Sekunder [4]. Kota Malang memiliki panjang total jalan sebesar 663,34 KM, yang terbagi menjadi jalan Arteri Primer sepanjang 11,82 km, Arteri Sekunder sepanjang 15,94 km, Kolektor Primer sebesar 8,16 km, Kolektor Sekunder sebesar 27,09 km, Lokal Primer sepanjang 9,66 km, dan Lokal Sekunder sepanjang 590,67 km [4]. Titik kemacetan paling parah pada pintu masuk kota Malang berada di pertigaan Karanglo yang termasuk jalur Arteri Primer. Sedangkan pertigaan Karanglo yang menjadi objek penelitian ini berada pada jalan Malang Kabupaten. Yang mana pertumbuhan jalan Kabupaten Malang yang diantaranya jalan Negara sepanjang 115,63 km (1%), jalan provinsi 114,93 km (1%), jalan Kabupaten 1.668,76 km (19%), dan jalan desa 6.907,90 km (79%), dan rata – rata pertumbuhan jalan di Kabupaten Malang hanya tumbuh 1% tiap tahun, berbanding dengan pertumbuhannya yang tumbuh 3,6% pertahun [bappekab.] Di pertigaan Karanglo ini arus semua kendaraan jadi satu, bayangkan saja, arus kendaraan yang ingin ke luar dari kota Malang dan dari luar kota Malang ke kota Malang dan dari luar kota Malang ingin ke kota Batu, arusnya bertemu jadi 1 di pertigaan Karanglo ini. Betapa macetnya jalan tersebut jika semua ingin berbagi jalan dalam 1 jalan yang sudah melebihi kapasitasnya. Pemerintah Malang sudah merencanakan solusi jangka panjang untuk kemacetan yang terjadi di pintu kota Malang ini, yaitu membangun jalan alternatif Lawang – Batu, dengan membangun jalur alternative Lawang – Batu

kedepannya arus jalan sudah tidak lagi menjadi satu bertemu di pertigaan karanglo, tetapi wisatawan yang dari luar kota malang ingin ke kota batu langsung bisa menuju kota batu tanpa harus melewati pertigaan karanglo. Arus kendaraan yang berkurang menjadi dampak positif dari kemacetan yang terjadi.

Tetapi membangun sebuah jalan baru membutuhkan perencanaan yang baik juga, dan dana yang tidak sedikit. Perlu banyak survey dan pertimbangan dalam segi kelayakan ekonomi maupun dari segi teknisnya. Pada 2013 proyek jalan ini sudah mulai direalisasikan, tetapi butuh waktu yang lama agar proyek jalan tembus lawang batu ini selesai. Sampai proyek ini belum rampung kemacetan parah masih terjadi. Maka dari itu dalam penelitian ini akan diusulkan solusi jangka pendek dalam menangani arus lalu lintas yang berada pada jalan karanglo, agar kemacetan dapat terurai.

2. TINJAUN PUSTAKA

2.1. Penelitian Terkait

Penelitian terkait dengan prediksi kemacetan sudah banyak dilakukan, mulai dari menggunakan pendekatan *Neural Network*, *PSO*, *SVM*, *Naïve Baiyes* maupun *Hybird method*. Dari beberapa penelitian juga ada yang menggunakan *Random Forest* dan *Adaboost* tetapi variable yang digunakan tidak sama dengan variable yang akan digunakan pada penelitian ini, beberapa penelitian terkait diantaranya seperti di bawah ini.

Tao Li, Liu Seng dalam penelitian berjudul *Prediction For Short-term Traffic Flow Based On Optimized Wavelet Neural Network Model* [5] memprediksi arus lalu lintas berjangka pendek dan meningkatkan akurasi prediksi menggunakan model algoritma *Wavelet Neural Network*, dengan cara mengoptimasi *Wavelet Neural Network* dengan memodifikasi algoritma menggunakan *Particle Swarm Optimization*. Hasil dari penelitian didapatkan dengan menggunakan hanya *Neural Network* saja dalam memprediksi arus lalu lintas diperoleh *error rate* sebesar 0.459776, menggunakan *Swarm Optimization* dan *Neural Network* mendapatkan *error rate* sebesar 0.372171, dan menggunakan *Particle Swarm Optimization* dengan *Wavelet Neural Network* mendapatkan hasil *error rate* sebesar 0.244689. Peneliti mendapatkan prediksi arus lalu lintas menggunakan *Wavelet Neural Network* dan dioptimasi menggunakan *Particle Swarm Optimization* lebih baik daripada menggunakan *NN* dan *Sawrm Optimization*.

Yan Wang, Yanyan Chen pada penelitian yang berjudul *Short-Term Traffic Flow Prediction By A Sugeno Fuzzy System Based on Gaussian Mixtures Models* [7], memprediksi arus lalu lintas menggunakan system inferensi Fuzzy Sugeno, yang ruang inputnya menggunakan model campuran Gaussian dan parameter yang diestimasi dengan metode *Least Square Estimation*. Hasil penelitian yang dilakukan oleh Yan Wang dan Yanyan Chen mendapatkan Mean Absolute Percentage Error (MAPE) dengan menggunakan *Fuzzy C- Means & Fuzzy Inference System (FCM-FIS)* sebesar 1.4541. Sedangkan menggunakan *Gaussian Mixture Models & Fuzzy Inference System (GMM-FIS)* mendapatkan MAPE sebesar 1.4751. Dan perhitungan menggunakan *Normalized Root Mean Square Error (NRMSE)* pada *FCM-FIS* didapatkan 0.0300, sedangkan pada *GMM-FIS* sebesar 0.0273.

Vedat TOPUZ, pada penelitian yang berjudul *Hourly Traffic Flow Predictions by Different ANN Models* [6], membandingkan beberapa model ANN untuk memprediksi arus lalu lintas tiap jam. Model ANN yang digunakan antara lain ANN dengan *Multi Layer Perceptron (MLP)*, kemudian model ANN dengan *Radial Basis Function Network (RBF)*, model ANN dengan *Elman Recurrent Neural Network (ERNN)*, dan model *Non-Linear Auto Regressive*, dan model *eXogenous input (NARX)*. Pada penelitian Vedat TOPUZ didapatkan hasil dengan menggunakan beberapa model ANN yang diusulkan dapat diketahui dari hasil perbandingan beberapa model ANN yang di . Hasilnya dapat diketahui ANN dengan *MLP* mendapatkan hasil eksperimen dengan pengukuran *RMSE* pada traning sebesar 0.0032 dan pada *testing* sebesar 0.0054. Pada model ANN dengan *ERN* didapatkan hasil pengukuran dengan *RMSE* pada *training* data sebesar 0.0556 dan pada *testing* data *RMSE* sebesar 0.0507. Dan pada model ANN dengan *RBF* didapatkan pengukuran pada traning data *RMSE* sebesar 0.0066 dan pada data *testing RMSE* yang

didapat sebesar 0.982. Pada model terakhir ANN dengan NARX didapatkan pengukuran hasil RMSE pada *training* data sebesar 0.0083 dan pada *testing* data sebesar 0.0127. Dapat disimpulkan pada penelitian Vedat TOPUS bahwa menggunakan ANN dengan beberapa model dapat digunakan untuk memprediksi arus lalu lintas dengan efisien.

Sucianna Ghadati Rabiha dan Stefanus sentosa dalam penelitiannya yang berjudul *Prediksi Data Arus Lalu Lintas Jangka Pendek Menggunakan Optimasi Neural Network Berbasis Genetik Algorithm* [9], menggunakan NN dan GA dalam memprediksi arus lalu lintas. Dalam penelitiannya membandingkan hasil prediksi menggunakan NN dan GA dengan NN Backpropagation tanpa menggunakan *Genetic Algorithm (GA)*, hasilnya prediksi arus lalu lintas dengan menggunakan *Neural Networks* berbasis *Genetic Algorithm* lebih kecil tingkat RMSE nya yaitu sebesar 106.016 dibandingkan dengan hanya menggunakan *Neural Network* saja yang mempunyai RMSE sebesar 108,780, terlihat penurunan RMSE sebesar 2,54% daripada hanya menggunakan Backpropagation *Neural Network*.

Poo Kuan Hoong, Ian K.T. Tan, Ong Kok Chien, Choo-Yee Ting [10] dalam penelitiannya yang berjudul *Road Traffic Prediction Using Bayesian Networks*, menggunakan service dari web cuaca dan tweet dari pengguna twitter untuk memprediksi keadaan arus lalu lintas, penelitian membandingkan 3 model Bayesian diantaranya yaitu *Bayesian Networks* disimbolkan dengan *M1*, model *Naïve Bayesian Networks M2*, dan model *Parameter-learning Bayesian Networks M3*. Hasilnya model *Parameter-learning Bayesian Networks M3* lebih akurat untuk memprediksi arus lalu lintas daripada menggunakan model *M2* dan *M3*.

Penelitian selanjutnya yang berkaitan dengan prediksi arus lalu lintas diantaranya dari Dr. Md Saiful Alam Siddquee, Hamid Uz Zaman, yang berjudul *Prediction of Hourly Traffic From Short Counts Using Artificial Neural Network and Support Vector Machine* [11]. Dalam penelitiannya Dr. Md Saiful menafsirkan memprediksi *missing data* yang digunakan dalam memprediksi arus lalu lintas menggunakan *Artificial Neural Network* maupun yang menggunakan *Support Vector Machine*. Dalam penelitiannya didapatkan hasil model *Support Vector Machine (SVM)* lebih akurat dalam memprediksi arus lalu lintas maupun dalam penanganan hal kehilangan data yang diolah daripada menggunakan model *Artificial Neural Network (ANN)*.

Penelitian terkait dengan arus lalu lintas selanjutnya dari Yunlon Huang yang berjudul *Traffic Flow Forecasting Based on Wavelet Neural Network and Support Vector Machine* [12]. Dalam penelitiannya menggunakan dua model algoritma yaitu *Wavelet Neural Network (WNN)* dan *Support Vector Machine (SVM)*, dijelaskan untuk memperbaiki nilai *peak* dari fungsi non linear digunakan model WNN karena jika menggunakan NN biasa nilai *peak* lebih tinggi daripada menggunakan WNN, setelah melakukan prediksi menggunakan WNN peneliti Yunlon Huang menggunakan SVM untuk meningkatkan nilai akurasi, hasilnya model WNN dengan SVM nilai akurasinya lebih tinggi daripada menggunakan NN biasa dalam memprediksi *Traffic Flow* atau arus lintas.

2.2. Landasan Teori

Machine Learning merupakan bagian dari Computer Science, pada mulanya berkembang dari pengenalan pola dan teori komputasi dalam kecerdasan buatan. Machine learning mengeksplorasi berdasarkan pembelajaran dan membangun sebuah algoritma yang dapat belajar (learn) secara mandiri untuk digunakan dalam memprediksi data. Algoritma tersebut berjalan dengan membangun sebuah model atau contoh masukan untuk memprediksi data maupun membuat keputusan. Machine Learning dan ilmu statistik sangat erat keterkaitannya, suatu ide dalam machine learning berawal dari prinsip – prinsip metodologis ke alat teoritis, dan memiliki sejarah keterkaitan yang erat dengan ilmu statistik.

Sebuah pembelajar atau biasa disebut dengan “*learner*” dimaksudkan untuk menggeneralisasi dari sebuah experience [13]. Generalisasi dalam konteks ini adalah kemampuan dari sebuah machine learning untuk melakukan secara akurat untuk memprediksi atau mempelajari secara mandiri setelah machine learning diberikan sebuah tugas dalam suatu rangkaian learning *dataset*. Contoh dari *training dataset* bisa didasarkan pada beberapa kemungkinan dan umumnya tidak diketahui (dianggap memiliki ruang

kejadian) dan *learner* secara otomatis membangun sebuah model umum tentang ruang ini, yang memungkinkan untuk menghasilkan sebuah prediksi yang cukup akurat dalam suatu kasus yang diberikan.

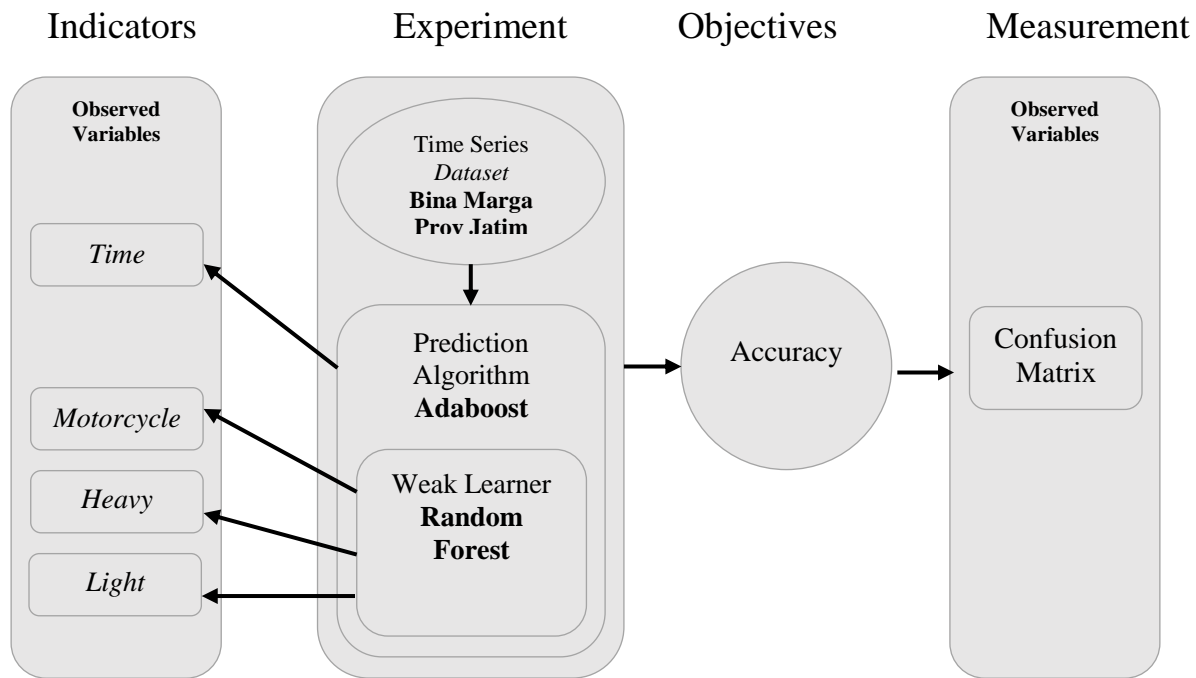
Data mining merupakan salah satu disiplin ilmu yang mempelajari metode dan mengekstrak pengetahuan atau menemukan pola dari suatu data. Data sendiri merupakan suatu fakta yang terekam dan tidak mempunyai arti. Dan pengetahuan merupakan suatu pola, aturan atau model yang muncul dari data, sehingga *Data mining* disebut dengan *Knowledge Discovery in Database (KDD)* [14] Transformasi data dalam *Data mining* bisa digambarkan sebagai berikut : Data -Informasi - Pengetahuan.

Dalam proses KDD bisa dibagi menjadi berikut [15]:

- a. Seleksi
- b. Pre – processing
- c. Transformasi
- d. *Data mining*
- e. Interpretasi / evaluasi

Dalam *Data mining* sebelum dilakukan penggalian data, terlebih dahulu *dataset* yang digunakan harus memasuki fase *Pre-processing*. Dalam fase *Pre-processing* data terlebih dahulu dinormalisasi agar nanti implementasi dari algoritma dapat berjalan baik. Kemudian *dataset* yang digunakan harus banyak atau besar sehingga tingkat pola yang dihasilkan semakin baik. *Pre-processing* sangat berguna untuk menganalisa *dataset* yang multi variat, target yang ditentukan kemudian dibersihkan dahulu. Pembersihan data menghilangkan *noise* dan *missing data*.

2.3. Model Eksperimen



Gambar 1. Model Eksperimen

3. METODE PENELITIAN

3.1. Pengumpulan Data

Dalam penelitian ini untuk mendapatkan *dataset* yang diperlukan maka dari itu digunakan beberapa metode pengumpulan data. *Dataset* yang digunakan dalam penelitian ini adalah data arus lalu lintas yang bersifat time series dan didapatkan dari Dinas Bina Marga Provinsi Jawa Timur periode 2014. *Dataset* yang diperoleh sebanyak 769 sheet atau 769 *record*, diibaratkan satu *record* sheet mewakili beberapa atribut. Dimensi *dataset* yang didapat sebanyak 769 row dan 12 *Column*. Setiap row mewakili value dari *column*. Dan *column* bisa dijadikan atribut yang akan digunakan, dan value terdapat pada row. Seperti contohnya dalam *column*/atribut *Pukul* terdapat value pada row yaitu 06.00 – 06.15, *Column*/Atribut *Kendaraan ringan (LV)* terdapat value pada row yaitu 135, *column*/atribut *Kendaraan berat (HV)* terdapat value row yaitu seperti 76, *column*/atribut *Sepeda motor* terdapat value pada row yaitu seperti 809, *column*/atribut *Kendaraan tak bermotor* terdapat value seperti 17, *column*/atribut *Jumlah* terdapat value seperti 1020, *column*/atribut *Volume* terdapat value seperti 1758,6, dan *column*/atribut *Arus* terdapat value pada row seperti 439,65

3.2. Pre-processing

Data yang didapat masih perlu dilakukan normalisasi terlebih dahulu. Tahapan normalisasi dalam *Data mining* dilakukan pada fase *Pre-processing*. Adapun atribut dalam *dataset* yang akan digunakan tidak semuanya dipakai. Seperti yang dijelaskan di atas ada 8 atribut dalam *dataset* yang dicantumkan, tetapi dalam tahap normalisasi ini peneliti tidak menyertakan atribut *Kendaraan tak bermotor (UM)* dikarenakan atribut tersebut tidak termasuk dalam bagian dari arus lalu lintas.

3.3. Eksperimen

Ide pokok dari *Boosting* algoritma adalah mengkombinasi beberapa model *accurate classifier* (disebut dengan *weak classifier*) menjadi hanya satu *classifier*. *Weak classifier* (klasifikasi lemah) *training* secara konseptual dan berurutan, sebagian dari *classifier training* berdasarkan pada sample yang diambil. Yang mana sangat sulit untuk mengklasifikasi dengan memprediksi *classifier* yang lemah. *Boosting algorithm* memberikan input ke pada data *training* set seperti contoh $S = \{(x_1, y_1), \dots, (x_m, y_m)\}$ dimana setiap inisiasi dari x_i adalah vector dari nilai atribut yang dimiliki oleh sebuah domain atau ruang misalkan X , dan setiap label y_i adalah label class yang berasosiasi dengan x_i dan memiliki ruang terbatas dari label Y . dalam kasus ini peneliti hanya 29ecto pada masalah klasifikasi biner saja, seperti $Y = \{-1, +1\}$. Klasifikasi akhir, h , dibangun berdasarkan berat vote dari setiap individual *classifier* h_1, h_2, \dots, h_m . Setiap *classifier* ditimbang berdasarkan akurasi, untuk pendistribusian p_1 dilakukan pelatihan data terlebih dahulu. Adapun pseudo code dari algoritma *adaboost* sebagai berikut :

- a. Memasukkan : a set S , dari label m , contohnya :
- b. $S = ((x_i, y_i), i = (1, 2, \dots, m))$ dengan label Y
- c. Learn (*Learning Algoritim Random Forest*)
- d. A konstanta dari L
 - 1) Inisialisasi dari semua i : $w_i(i) = \frac{1}{m}$ menginisialisasi semua beratnya.
 - 2) Untuk $l=1$ lakukan sampai L
 - 3) Untuk semua i : $p_l(i) = \frac{w_i(i)}{\sum_i w_i(i)}$ mengkomputasi normalisasi berat.
 - 4) $h_l = \text{Learn}(S, p_l)$ disebut pembelajaran dengan normalisasi berat.
 - 5) $\epsilon_l = \sum_i p_l(i)[h_l(x_i) \neq y_i]$ disebut pembelajaran dengan normalisasi berat.
 - 6) *if* $\epsilon_l > \frac{1}{2}$ maka mengkalkulasi *error* dari h_l .
 - 7) $L = l - 1$
 - 8) Maka ke 12
 - 9) $\beta_l = \frac{\epsilon_l}{1 - \epsilon_l}$

10) untuk semua i : $w_{l+1}(i) = w_l(i)\beta_l^{1-(h_l(x_i \neq y_i))}$ mengkomputasi berat baru.

11) Akhiri

12) Hasilnya $h_{akhir}(x) = \operatorname{argmax}_{y \in Y} \sum_{l=1}^L \left(\log \frac{1}{\beta_l} \right) [h_l(x) = y]$

Metode *Random Forest* mulai diperkenalkan oleh Breiman dan merupakan sebuah pohon klasifikasi gabungan. Dapat dijelaskan dalam pembentukan metode *Random Forest* bisa digambarkan jika gugus data *training* yang digunakan memiliki nilai n dan terdiri dari d sebagai peubah penjelas (Predictor). Adapun proses dalam pendugaan dan penyusunan dari metode *Random Forest* bisa dijelaskan sebagai berikut :

- a. (proses bootstrap) menarik sebuah contoh acak dengan pemulihan berukuran nilai n dari sebuah gugus data *training* yang digunakan.
 - 1) (proses random sub-setting) menyusun pohon berdasarkan data yang diambil, tetapi pada setiap proses pemisahan selalu memilih acak $m < d$ artinya peubah dan penjelas kemudian lakukan pemisahan dan pilih yang terbaik dari pemisah tersebut.
 - 2) Ulangi proses a – b di atas sebanyak k kali sehingga didapatkan k buah pohon secara acak.
- b. Dari hasil point c didapatkan k buah pohon tersebut (di umpamakan menggunakan *majority vote* untuk permasalahan menggunakan klasifikasi, atau menggunakan perhitungan average untuk masalah berbasis regresi).

Metode RF ini dilihat secara sepintas sama dengan metode *bagging*, dapat diketahui setiap kali dalam pembentukan suatu pohon, kandidat dari peubah dan penjelas yang digunakan untuk meakukan pemisahan bukanlah seluruh peubah yang terlihat tetapi hanya sebagian dari hasil pemilihan secara random hasil dari proses disamping adalah kumpulan pohon tunggal (*ensemble tree*) dengan ukuran dan bentuk tidak sama. Hasilnya merupakan kumpulan pohon tunggal yang memiliki korelasi kecil antar pohonnya. Akibat dari korelasi yang kecil tersebut maka berbagai macam dugaan atau prediksi dapat terbentuk tetapi memiliki presentasi yang kecil daripada hasil prediksi dari metode *bagging*.

Random Forest terdiri dari 3 prediktor, kombinasi dari predictor yang digunakan oleh setiap pohon tergantung pada nilai – nilai dari vector yang diambil secara random (acak) dan dengan distribusi yang sama untuk semua *tree* pada forest. Kesalahan pada generalisasi pada forest menjadi satu dan mengakibatkan adanya batas antara jumlah pohon menjadi besar. Kesalahan dalam mengklasifikasi pohon sangat tergantung pada kekuatan setiap pohon tunggalnya dan korelasi atau hubungan antar pohon tersebut. Estimasi internal dari RF dapat memonitor kesalahan, kekuatan, dan korelasi ini digunakan untuk menunjukkan respon terhadap jumlah fitur yang digunakan untuk mode splitting. Elemen utama dalam semua prosedur ini adalah setiap pohon k , sebuah θ_k vector diambil secara acak dan independensi dari vector acak $\theta_1 \dots \theta_{k-1}$. Tetapi untuk pendistribusiannya harus secara merata. Dan pohon bisa tumbuh dengan menggunakan *training* set dan θ_k , kemudian menghasilkan sebuah *classifier* $h(x, \theta_k)$, dimana x adalah input dari random vector. Misalkan hasil diatas mempunyai sampel 30 vector acak θ yang mempunyai ukuran N yang dipilih dari penggantian dari contoh N dari data *training* set. Setelah sejumlah pohon dihasilkan, maka selanjutnya memilih (vote) class yang paling populer. Adapun pseudo-code dari metode *Random Forest* bisa dilihat sebagai berikut :

- a. Memilih nilai dari pohon K yang akan dihasilkan.
- b. For $k=1$ maka lakukan sebanyak K .
- c. Maka dihasilkan sebuah Vektor θ_k .
- d. Membuat *tree* $h = (x, \theta_k)$ menggunakan algoritma *decision tree* model apapun.
- e. Memilih salah satu pohon untuk direkomendasikan diambil dari class paling populer X .
- f. Class X memprediksi dengan menseleksi class dengan voting secara maksimal.
- g. Kembali ke hypothesis h_l
- h. End For.

4. HASIL PENELITIAN DAN PEMBAHASAN

4.1. Pre-processing

Agar lebih ringkas *dataset* dinormalisasi dan hanya mencantumkan atribut dan value yang digunakan untuk memprediksi. Atribut yang digunakan dalam memprediksi adalah sebagai berikut : Pukul, LV, HV, dan MC. Adapapun atribut yang lain seperti UM tidak diikutkan karena dianggap bukan sebagai penghambat lalu lintas dan tidak bermotor. Maka untuk lebih detail dalam normalisasi data dan hasil normalisasi bisa dilihat dalam tabel di bawah ini.

Tabel 1. Hasil Normalisasi

Ha ri	Pukul	LV	H V	MC	LV	HV	MC	Arus / 15 menit	Volum e / Jam	Kapasita s jalan / jam	LOS	Y
					E	E	E					
					1	1,3	0,25					
6	06.00.00	166	63	546	166	81,9	136,5	384,4	1537,6	1735,992	0,885718367	-1
6	06.15.00	146	54	578	146	70,2	144,5	360,7	1442,8	1735,992	0,831109821	-1
6	06.30.00	190	68	762	190	88,4	190,5	468,9	1875,6	1735,992	1,080419725	1
6	06.45.00	210	80	702	210	104	175,5	489,5	1958	1735,992	1,127885382	1
6	07.00.00	194	90	789	194	117	197,25	508,25	2033	1735,992	1,171088346	1
6	07.15.00	202	83	700	202	107,9	175	484,9	1939,6	1735,992	1,117286255	1
6	07.30.00	198	65	711	198	84,5	177,75	460,25	1841	1735,992	1,060488758	1

4.2. Test dan Training Data

- a. Import *dataset* dari table matrix yang sudah dibuat pada saat konversi data.
- b. Menentukan sample *training dataset* $S = ((x_i, y_i), i = (1, 2, \dots, m))$ dengan label *Y*

S = sample

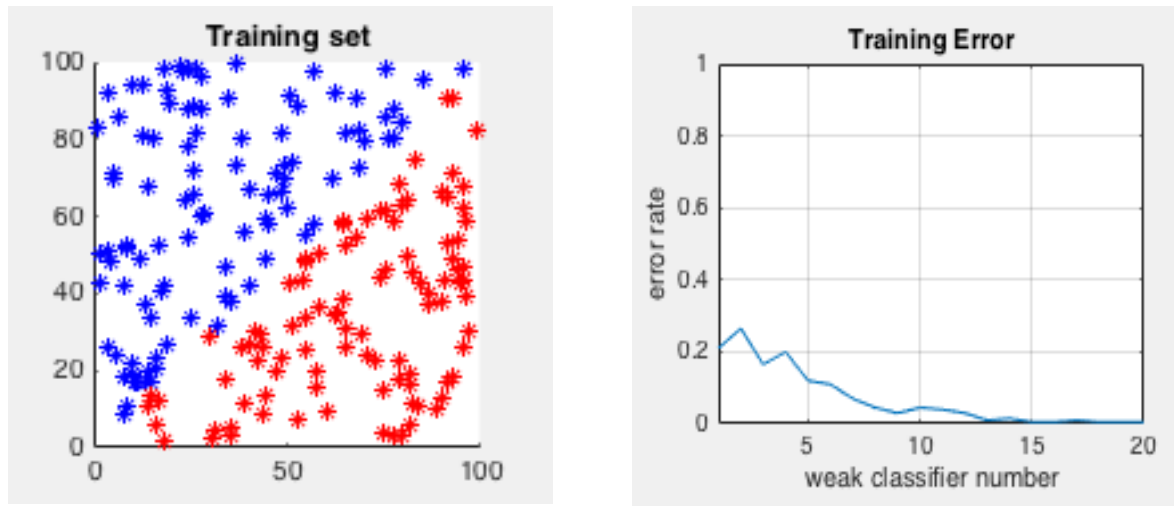
x_i = atribut domain

y_i = class

i = iteration

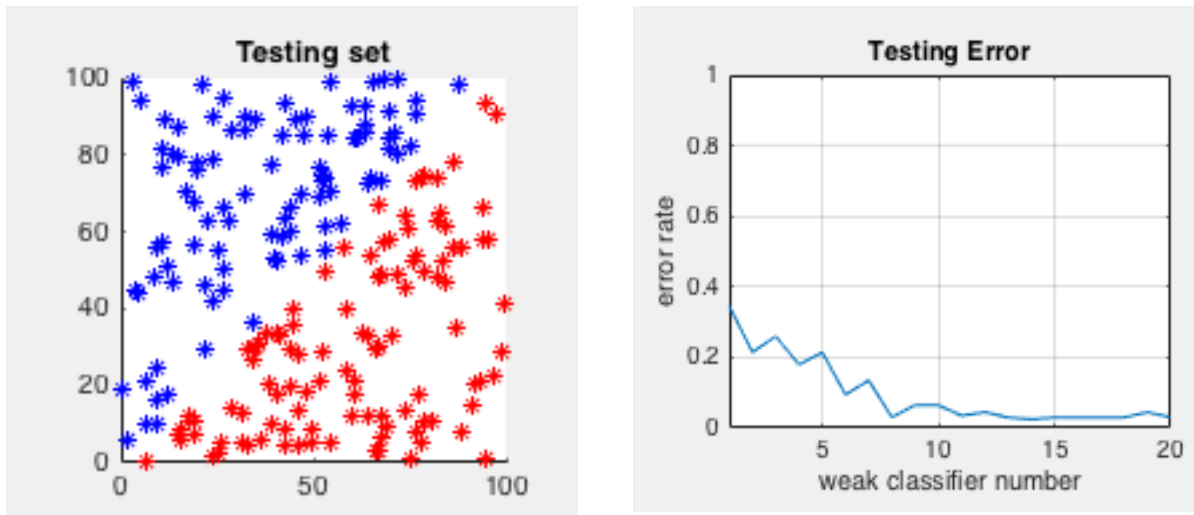
m = label *training*

Train data digunakan sebanyak 90 % dari *dataset*, dan dimulai dari kolom 1 sampai 8, sampai kolom terakhir dari 90% *dataset*.



Gambar 2. Result Train Data

Test data digunakan 10% dari keseluruhan *dataset*.



Gambar 3. Result Test Data

4.3. Weak Learner Random Forest

Membuat *tree* $h = (x, \theta_k)$ menggunakan algoritma *Ensemble decision tree*

$h = tree$

$x = predictor$

$\theta_k = \text{banyaknya nilai tree}$

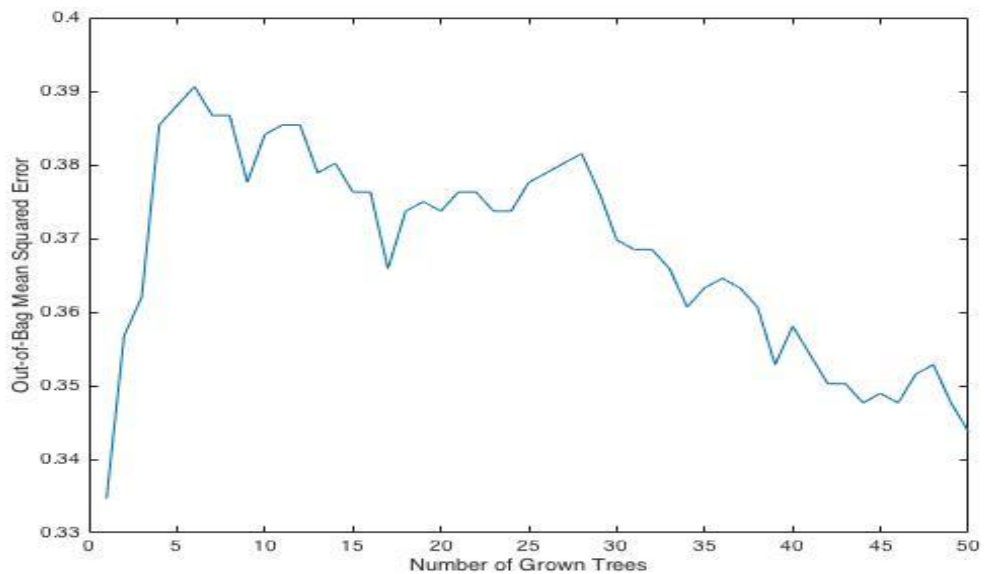
```
TreeBagger
Ensemble with 50 bagged decision trees:
  Training X:      [768x2]
  Training Y:      [768x1]
  Method:          classification
  NumPredictors:   2
  NumPredictorsToSample: 2
  MinLeafSize:     1
  InBagFraction:   1
```

```

SampleWithReplacement: 1
ComputeOOBPrediction: 1
ComputeOOBPredictorImportance: 0
Proximity: []
ClassNames: '-1' '1'
    
```

Gambar 4. Ensemble decision tree Bagger

Jika diilustrasikan hasil dari algoritma *ensemble decision tree* bisa dilihat pada gambar 4.10 di bawah ini :



Gambar 5. Illustrasi Tree Bagger

Seperti pada penjelasan pada bab III di pseudo code *Random Forest* dijelaskan untuk membangun sebuah *Random Forest* bisa menggunakan algoritma *decision tree*, dan pada *Eksperimentini* menggunakan ensemble *tree* bagger, karena dengan menggunakan *tree* bagger bisa meminimalisir kesalahan dalam klasifikasi.

4.4. Menentukan Weight

Setelah melakukan *training weak classifier* pada 4.1.2.2 maka selanjutnya menentukan bobot / weight dari weak *learner* yang dihasilkan oleh random forest, adapun untuk menentukan weight untuk algoritma *adaboost* sebagai berikut.

$$i: w_i(i) = \frac{1}{m}$$

$i = \text{atribut}$

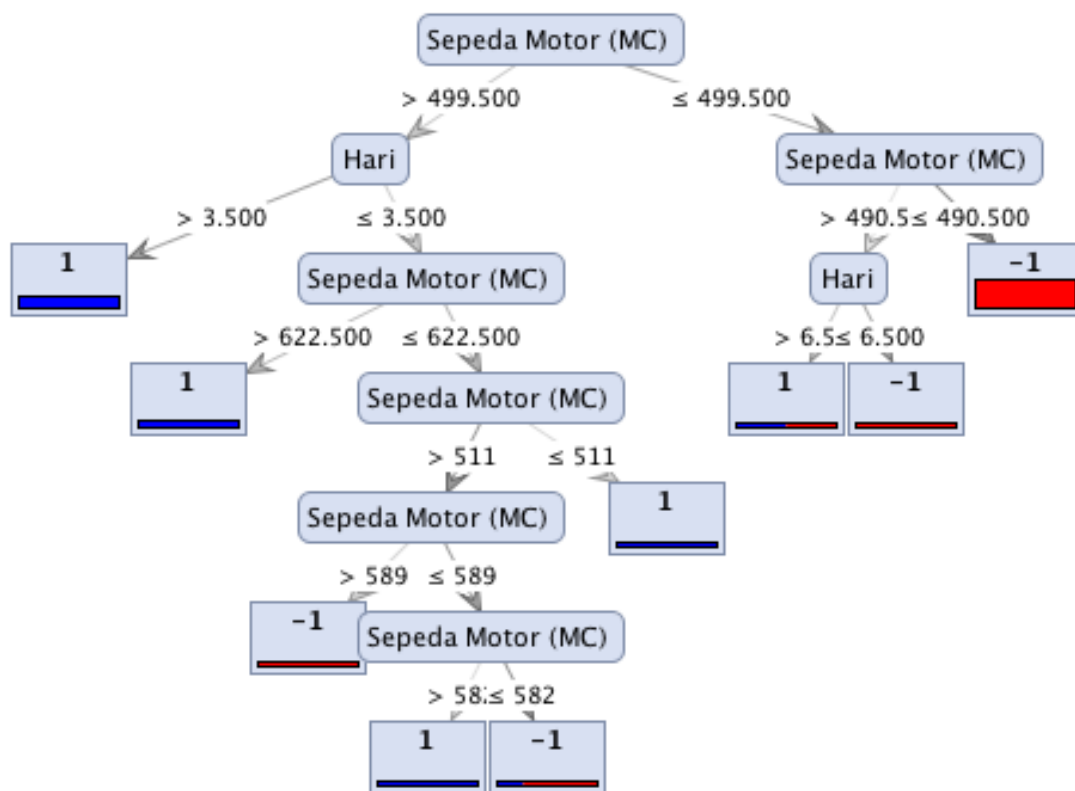
$w = \text{bobot}$

$m = \text{sample}$

4.5. Perbandingan Hasil Eksperimen

Dari hasil pengolahan algoritma *Adaboost* untuk *training dataset* dengan menggunakan hasil weight yang terbaik didapat hasil sebagai berikut :

$$\epsilon_t = \sum_i p_i(i)[h_t(x_i \neq y_i)]$$

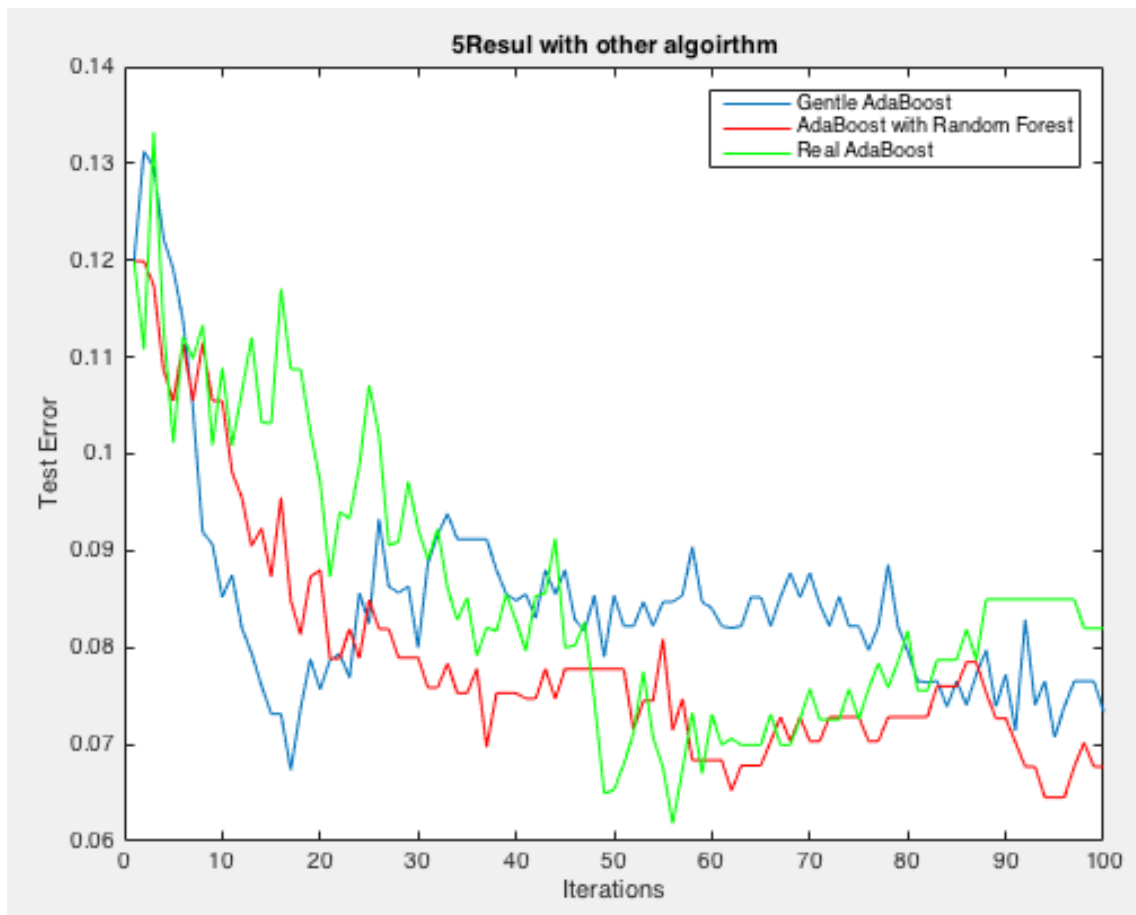


Gambar 6. Eksperimen 4.4.1 : *Adaboost model Weight = 3.916*

Hasil *training* dan *test* menggunakan *Adaboost* dan *Random Forest* bisa diketahui dari eksperimen 4.4.2, yang hasilnya menunjukkan tingkat *error*-nya lebih sedikit dari pada menggunakan *Adaboost* versi asli. Untuk lebih detail hasil perbandingan dengan algoritma lain bisa dilihat di eksperimen 4.4.3

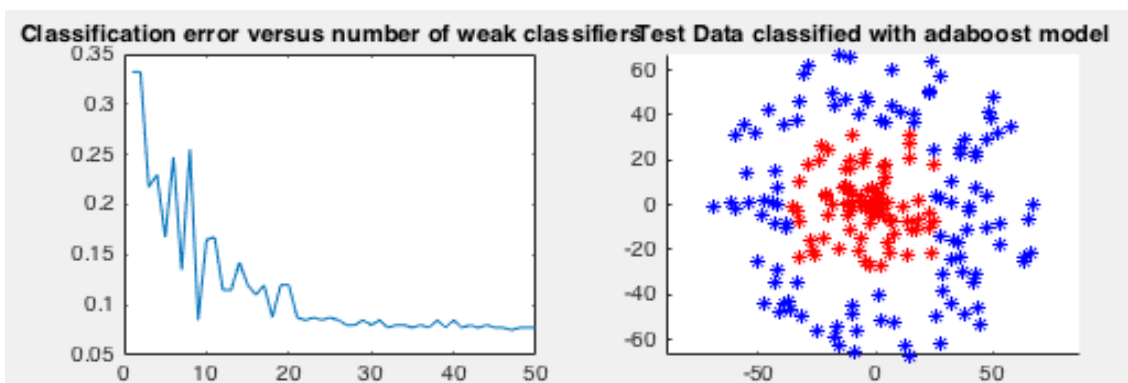
Sebagai algoritma pembanding digunakan algoritma *adaboost* asli yang disebut dengan *Real Adaboost* dan Algoritma pembanding kedua yang lebih baik dari *Real Adaboost* yaitu *Gentle Adaboost* yang sama – sama hanya menggunakan weak *classifier* asli dari algoritma tersebut, kedua algoritma tersebut akan dijadikan pembanding dengan algoritma yang digunakan dalam penelitian ini yaitu *Adaboost* dengan

Random Forest, hasilnya bisa dilihat di bawah ini :



Gambar 7. Eksperimen 4.4.3 : Perbandingan hasil Algoritma

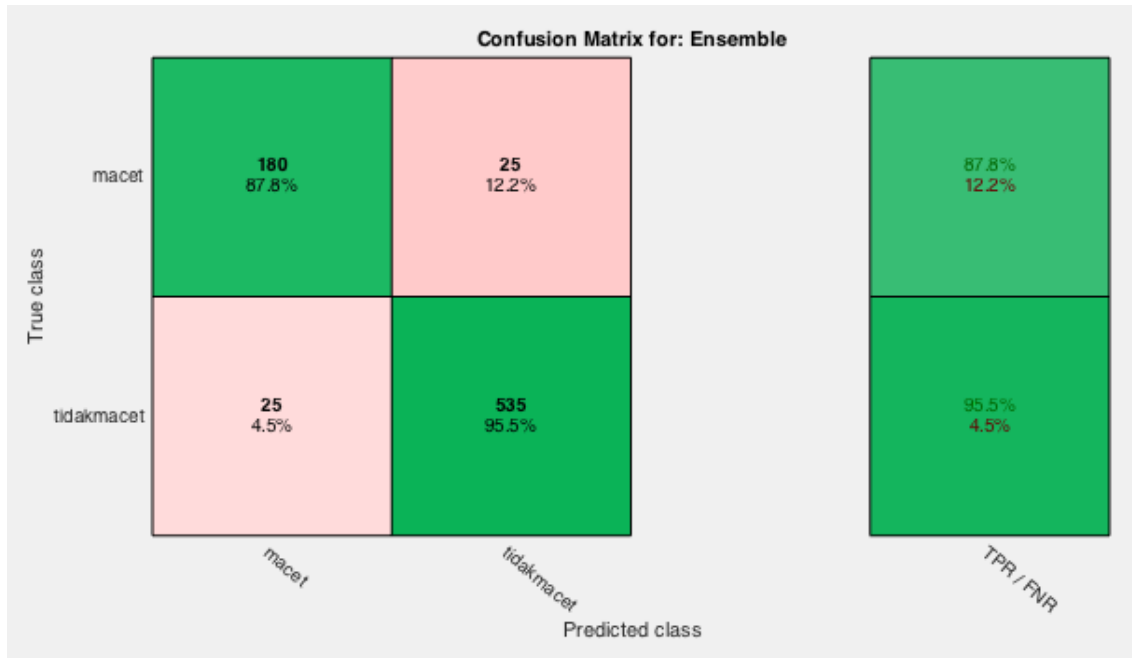
Kalkulasi *error* dari h_1 . Untuk mengkalkulasi tingkat *error* dari learning *training* dari *adaboost* digunakan rumus *if* $\epsilon_l > 1/2$



Gambar 8. Eksperimen 4.4.4 : Error h_1

Dari eksperimen dihasilkan tingkat *error* sebesar 0,0629 artinya semakin mendekati 0 semakin kecil pula tingkat *error*nya.

Evaluasi menggunakan confusion matrix dengan hasil sebagai berikut.



Gambar 9. Eksperimen 4.4.5 : Confusion Matrix

- Nilai True Positive untuk macet sebesar 180 sedangkan yang tidak macet sebesar 535. Selain nilai True Positive disebut false negative. Untuk nilai false negative untuk macet sebesar 25 dan untuk tidak macet sebesar 25.
- Presisi untuk macet adalah 180 dibagi 180 + 25 hasilnya sebesar 12,2% atau 87,8%
- Class recall untuk macet adalah 180 dibagi 180 + 25 dan hasilnya adalah 4,5 atau 97,83%
- Akurasi secara keseluruhan adalah $180 + 535 / 180 + 535 + 25 + 25 = 87,8$ atau 87,8%

5. KESIMPULAN DAN SARAN

5.1 KESIMPULAN

Dari hasil analisa metode *Adaboost* dengan weak learner *Random Forest* untuk memprediksi arus lalu lintas jangka pendek maka didapat kesimpulan sebagai berikut :

- Metode *Adaboost* dan *Random Forest* sebagai weak learner untuk memprediksi arus lalu lintas jangka pendek mempunyai akurasi sebesar 87,8 %.
- Metode *Adaboost* dengan weak learner *Random Forest* terbukti bisa meningkatkan akurasi dalam pengklasifikasi di dalam weak learner *Adaboost*.
- Metdode *Adaboost* dengan weak learner *Random Forest* terbukti bisa sebagai alternatif dalam memprediksi data time series untuk arus lalu lintas jangka pendek.

1.2 Saran

Dari hasil penelitian ini masih banyak kekurangan dalam cara bereksperimen dengan metode *Adaboost* dan *Random Forest* diantaranya yaitu :

- a. Penelitian ini masih belum sempurna dan peneliti hanya mengimplementasikan penelitian hanya pada *dataset* yang bersifat private, harapan ke depan bisa diimplementasikan pada *dataset* yang memang benar bersifat public.
- b. Membandingkan hasil penelitian dengan metode lain yang tingkat akurasi lebih baik.
- c. Pada penelitian selanjutnya bisa diharapkan penerapan *Adaboost* dengan *Random Forest* bisa ditingkatkan lagi akurasinya dengan beberapa eksperimen dan parameter maupun *dataset* yang lebih banyak lagi.

ACKNOWLEDGEMENT

Ucapan terimakasih disampaikan kepada Bpk Heru Agus Santoso Ph.d dan Bpk Aris Marjuni yang senantiasa memberikan masukan dan bimbingan sehingga tesis ini bisa berjalan dengan baik. Kepada para pengajar dosen di lingkungan Udinus diantaranya Bpk Dr. Abdul Syukur, dan Bpk Purwanto yang selalu memberikan arahan tentang perjalanan kuliah saya.

PERNYATAAN ORIGINALITAS

“Saya menyatakan dan bertanggung jawab dengan sebenarnya bahwa artikel ini adalah hasil karya saya sendiri kecuali cuplikan dan ringkasan yang masing – masing telah saya jelaskan sumbernya”

Ahmad Rofiqul Muslikh – P31.2013.01510

DAFTAR PUSTAKA

- [1] Xerox Company, "Transportation Analytics," Xerox Corporation, 2014.
- [2] Badan Pusat Statistik, "Statistik Transportasi Darat," Subdirektorat Statistik Transportasi, Katalog BPS 2339-2746, 2014.
- [3] Badan Pusat Statistik, "Produk Domestik Regional Bruto Kabupaten / Kota di Indonesia 2010 - 2014," Badan Pusat Statistik, Katalog BPS 1907-8242, 2010 - 2014.
- [4] Bappeda Jatim, "Eksum Rencana Induk Jalan," BAPPEDA, Executive Summary 2012.
- [5] Tao Li and Liu Seng, *Prediction for Short-Term Traffic Flow Based on Optimized Wavelet Neural Network Model*, vol. 7, No 2.
- [6] Vedat TOPUZ, "Hourly Traffic Flow Prediction by Different ANN Models," The Marmara University, Istanbul, 978-953-307-100-8, 2010.
- [7] Yang Wang and Chen Yanyan, "Short-Term Traffic Flow Prediction by a Sugeno Fuzzy System Based on Gaussian Mixture Models," Beijing University of Technology, Beijing, 1992-8645, 2012.
- [8] Leo Breiman, "*Random Forest*," University of California, Berkeley, 2001.
- [9] Rabiha Sucianna Ghadati and Stefanus Santosa, "Prediksi Data Arus Lalu Lintas Jangka Pendek Menggunakan Optimasi *Neural Network* Berbasis Genetik *Algorithm*," *Jurnal Teknologi Infomasi*, vol. 9 Nomor 2, Oktober 2013.
- [10] Poo Kuan Hoong, Ian. K. T. Tan, Ong Kok Chien, and Choo-Ye Ting, "Road Traffic Prediction Using Bayesian Networks," Multimedia University, Cyberjaya, Malaysia,.
- [11] Dr. Md. Saiful Alam SIDDIQUEE and Hamid-Uz-ZAMAN, "Prediction of Hourly Traffic from Short Count Using Artificial *Neural Network* and Support Vector Machine," *Journal of Society for Transportation and Traffic Studies (JSTS)*, vol. 4 No 2, 2014.
- [12] Yunlon HUANG, "Traffic Flow Forecasting Based on *Wavelet Neural Network* and Support Vector

Machine," *Journal of Computational Information System*, vol. 8 No 8, 2012.

- [13] Christoper M. Bishop, *Pattern Recognition and Machine Learning*, Michael Jordan, Professor Jon Kleinberg, and Bernhard Scho lkopf, Eds. New York, United States of America: Springer, 2006.
- [14] Usama Fayyad, Gregory Piatetsky-Shapiro, and and Padhraic Smith, "From *Data mining to Knowledge Discovery in Databases*," New York University, New York, 1997.
- [15] Óscar Marbán, Gonzalo Mariscal, and Javier Segovia, *A Data mining & Knowledge Discovery Process Model.*: I-Tech Education and Publishin, 2009.
- [16] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu, "A Symbolic Representation of Time Series, with Implications for Streaming *Algorithms*," University of California, Riverside, 2003.
- [17] Lior Rokach and Oded Maimon., Danvers, United States of America: Scientific Publishing Co. Pte. Ltd, 2008, ch. 69.
- [18] Leo Breiman, "*Bagging* Predictors," University of California Berkeley, Berkeley, 2004.
- [19] Ludmila I. Kuncheva and Juan J. Rodriguez, "An Experimental Study on Rotation Forest Ensembles," University of Wales, Bangor, 459–468, 2007.