

Metode Hibrida FCM dan PSO-SVR untuk Prediksi Data Arus Lalu Lintas

Agri Kridanto^{*1}, Joko Lianto Buliali^{*2}

^{*}Jurusan Teknik Informatika, Institut Teknologi Sepuluh Nopember
Jalan Teknik Kimia ITS, Surabaya

¹agri.kridanto@gmail.com

²joko@cs.its.ac.id

Abstract — Traffic flow forecasting is one important part in Intelligent Transportation System. There are many methods had been developed for time series and traffic flow forecasting such as: Autoregressive Moving Average (ARIMA), Artificial Neural Network (ANN), and Support Vector Regression (SVR). SVR performance depend on kernel function and parameters of those kernel and data characteristic used in SVR as well. This research proposed hybrid method for traffic flow data clustering and forecasting. Fuzzy C-means is used in order to minimize the variance in whole dataset. Particle Swarm Optimization (PSO) is used in order to select the appropriate parameters for SVR. Experimental result shows the proposed method give MAPE below 4% in all test sites.

Keywords—fuzzy c-means, particle swarm optimization, prediksi data lalu lintas, support vector regression, time-series.

I. PENDAHULUAN

Tersedianya berbagai sensor dan detektor dapat memberikan kemudahan bagi para pengambil keputusan dan para peneliti untuk mendapatkan data lalu lintas pada suatu daerah. Data lalu lintas tersebut dapat berupa tingkat arus lalu lintas, kepadatan jalan, kecepatan kendaraan hingga waktu perjalanan yang dicatat dalam interval waktu tertentu dalam suatu periode waktu pengamatan. Data lalu lintas tersebut umumnya bersifat non linear dan non stasioner.

Prediksi data lalu lintas menjadi salah satu area riset dan aplikasi yang sangat penting dalam pengembangan Sistem Transportasi Cerdas. Ketersediaan data lalu lintas pada masa lampau sangat bermanfaat untuk mengembangkan model prediksi data lalu lintas. Pengembangan model prediksi data lalu lintas yang akurat dan *real time* menjadi salah satu tantangan utama dalam pengembangan model prediksi data lalu lintas[1]. Hal tersebut dapat memberikan manfaat bagi para pengambil keputusan untuk menentukan kebijakan transportasi dan deteksi dini untuk mengatasi kemacetan. Selain itu model prediksi data lalu lintas yang akurat dan *real time* dapat memberikan informasi kondisi lalu lintas

kepada para pengguna jalan. Metode prediksi data lalu lintas diharapkan dapat memberikan nilai deviasi yang minimal antara data sebenarnya dengan data yang diprediksi.

Prediksi data lalu lintas telah menjadi area riset yang sangat luas dengan dikembangkannya berbagai metode prediksi data lalu lintas baik dalam kategori statistik maupun *machine learning* [2]. Dalam kategori statistik, model Autoregressive Moving Average (ARMA) yang dikembangkan oleh Box dan Jenkins menjadi metode yang sangat fundamental dalam prediksi data *time-series*. Berbagai model telah dikembangkan untuk memperbaiki ARMA, seperti model: Autoregressive Moving Average (ARIMA), Kohonen-ARIMA (KARIMA), Seasonal ARIMA (SARIMA), dsb. Dalam kategori *machine learning*, metode Artificial Neural Network (ANN) dan Support Vector Regression (SVR) merupakan metode yang sering digunakan untuk prediksi data lalu lintas. Kedua metode tersebut merupakan metode *supervised learning* yang dapat dilatih untuk mendapatkan fungsi dari data masukan dan data keluaran yang merepresentasikan suatu label atau nilai yang akan diprediksi.

SVR sebagai metode pengembangan dari Support Vector Machine (SVM) memiliki performa yang handal dalam prediksi data *time series*. SVR telah sukses digunakan dalam berbagai area penelitian, seperti: prediksi data lalu lintas([2]–[6]), prediksi pergerakan harga saham[7], prediksi pancaran tenaga surya[8], prediksi data penjualan[9], dsb. Beberapa penelitian menunjukkan performa SVR lebih baik dibandingkan model ANN dalam prediksi data lalu lintas([2], [4], [5]) Namun kehandalan performa SVR sangat ditentukan oleh pemilihan parameter pada fungsi kernel yang digunakan serta karakteristik dari data yang digunakan dalam membangun SVR. Penggunaan metode *clustering* dapat memberikan karakteristik data yang lebih baik dan dapat meningkatkan performa dari metode SVR ([6], [8]).

Pada umumnya model prediksi data lalu lintas menggunakan keseluruhan dataset ataupun rentang waktu yang telah ditentukan. Salah satu kelemahan utama dalam

penggunaan keseluruhan dataset adalah karakteristik data yang sangat bervariasi dalam dataset tersebut. Pada penelitian ini diajukan sebuah metode hibrida dengan pengelompokan dan prediksi data lalu lintas. Metode Fuzzy C-Means (FCM) digunakan untuk mengelompokkan dataset berdasarkan informasi arus lalu lintas. Metode SVR digunakan sebagai metode prediksi yang dibangun untuk setiap kelompok yang dihasilkan. Metode PSO digunakan untuk proses pemilihan parameter pada SVR karena memiliki konsep yang sederhana dan efisien dalam mencapai solusi global.

II. PENELITIAN TERKAIT

Hong dkk. menggunakan metode Continuous Ant Colony Optimization (CACO) untuk melakukan pemilihan parameter SVR dengan fungsi kernel RBF[3]. Metode tersebut memiliki performa yang lebih baik dibandingkan model SARIMA. Pada penelitian yang lain Hong dkk. menggunakan metode Genetic Algorithm dan Simulated Annealing (GA-SA) untuk menentukan parameter SVR[6]. Metode yang diajukan memiliki performa yang lebih baik dibandingkan SARIMA, Back-Propagation Neural Network (BPNN), Holt Winter (HW), dan Seasonal Holt Winter (SHW).

Deng dkk. menggunakan algoritma Hybrid Chaotic Genetic Algorithm (CGA) untuk menentukan parameter pada SVR[5]. Hybrid CGA digunakan untuk mengatasi kelemahan GA yang kerap terjebak lokal optimum dan memiliki biaya komputasi yang tinggi. Metode yang diajukan memiliki performa yang lebih baik dibandingkan GA-SVR dan NN.

Fang dkk. menggunakan metode Improved Artificial Fish Swarm Algorithm (IAFSA) untuk menentukan parameter SVR[6]. Metode IAFSA diajukan untuk mengatasi masalah pada metode Grid Search yang memiliki waktu komputasi yang lama terutama untuk data yang besar. Selain itu, IAFSA diajukan untuk mengatasi kelemahan PSO dan GA yang sering terjebak lokal optima. Metode yang diajukan memiliki performa yang lebih baik dibanding metode Grid-SVR, GA-SVR, dan PSO-SVR.

Lippi dkk. melakukan perbandingan 9 metode prediksi data lalu lintas, antara lain : SARIMA-Kalman Filter, SVR, ANN, dsb[2]. Selain itu, Lippi dkk. juga mengajukan modifikasi fungsi kernel RBF dan kernel linear untuk kondisi musiman (*seasonal kernel*). Metode kombinasi SARIMA-Kalman Filter memiliki performa terbaik namun memiliki biaya komputasi yang sangat tinggi. Model SVR dengan *seasonal kernel* memberikan performa terbaik kedua diantara metode yang digunakan. Hasil perbandingan lainnya menunjukkan bahwa performa model SVR dengan kernel RBF memberikan performa yang lebih baik dari metode ANN dalam semua ujicoba yang digunakan.

Selain pemilihan parameter kernel, karakteristik dari data yang digunakan juga berpengaruh terhadap performa dari metode SVR. Lun dkk. menggunakan metode

Independent Component Analysis (ICA) untuk proses seleksi fitur dalam data penjualan produk IT[9]. Fitur yang telah didapatkan dikelompokkan dengan metode K-means. Model Prediksi SVR dibangun untuk setiap kelompok yang terbentuk. Metode yang diusulkan memiliki performa lebih baik dibandingkan metode SVR, Kmeans-SVR, dan ICA-GHSOM-SVR. Dong dkk menggunakan metode Self Organizing Maps (SOM) dan PSO-SVR untuk prediksi tenaga surya[8]. SOM digunakan untuk membagi dataset menjadi beberapa bagian berdasarkan korelasi antara data masukan dan keluaran. Data dalam setiap kelompok yang dihasilkan digunakan untuk membangun model prediksi PSO-SVR. Metode SOM-PSO-SVR memiliki performa yang handal dalam prediksi pancaran tenaga surya dan memiliki performa yang lebih baik dibandingkan model ARIMA, Random Walk, dan Exponential Smoothing.

Dalam topik pengelompokan data lalu lintas, Kianfar dan Edara menggunakan metode Kmeans, Gaussian Mixture Model (GMM), dan Hierarchical Clustering untuk membagi kondisi lalu lintas menjadi kondisi macet (*congested flow*) dan kondisi lancar (*free flow*) [10]. Atribut yang digunakan untuk proses pengelompokan adalah arus lalu lintas, rata-rata kecepatan dan *occupancy* serta setiap kombinasi dari atribut tersebut. Metode K-means dan Hierarkikal Clustering memberikan hasil pengelompokan yang lebih baik daripada GMM.

III. DASAR TEORI

A. Fuzzy C-Means (FCM)

FCM adalah metode pengelompokan data yang diusulkan oleh Bezdek dengan memanfaatkan konsep dari *fuzzy theory*[11]. Dalam FCM, setiap data memiliki derajat keanggotaan dalam setiap kelompok. Derajat keanggotaan tersebut memiliki nilai diantara 0 dan 1. Suatu data memiliki derajat keanggotaan sama dengan 0 jika data tersebut sama sekali tidak menjadi bagian dalam sebuah kelompok. Sementara nilai 1 menunjukkan bahwa data merupakan anggota sepenuhnya dalam suatu kelompok.

Hal tersebut merupakan perbedaan utama dengan algoritma '*hard clustering*', seperti K-means, dimana suatu data merupakan anggota suatu kelompok atau tidak. Seperti halnya dalam algoritma K-means, FCM memiliki titik pusat (*centroid*) kelompok serta menentukan derajat keanggotaan u_{ij} untuk setiap data i pada kelompok j . Algoritma FCM memiliki fungsi objektif dengan meminimalkan fungsi pada persamaan 1:

$$F = \sum_{i=1}^N \sum_{j=1}^c (u_{ij})^w \|x_i - c_j\|^2 \quad (1)$$

Fungsi objektif pada persamaan 1 digunakan untuk meminimalkan jarak antar data dalam kelompok yang sama (*intra-cluster*) sehingga suatu data memiliki kemiripan dengan data lain pada kelompok yang sama. Nilai w adalah nilai bobot yang diperkenalkan dalam algoritma FCM. Nilai $w > 1$ dan umumnya diberikan nilai 2.

Berikut ini merupakan tahapan dari algoritma FCM :

1. Tentukan banyak kelompok yang akan dibentuk.
2. Tentukan maksimum iterasi dan batas ambang perubahan nilai fungsi objektif.
3. Inisialisasi pusat kelompok secara acak, C_1, C_2, \dots, C_k .
4. Inisialisasi derajat keanggotaan, $\sum u_{ij} = 1$
5. Hitung titik pusat kelompok,

$$C_j = \frac{\sum_{i=1}^n (u_{ij})^w x_i}{\sum_{i=1}^n (u_{ij})^w} \quad (2)$$

dimana:

n = jumlah data dalam kelompok j

x_i = data ke- i

u_{ij} = derajat keanggotaan data i pada kelompok j

6. Perbarui derajat keanggotaan,

$$u_{ij} = \frac{D(x_i, c_j)^{\frac{1}{w-1}}}{\sum_{k=1}^C D(x_i, c_k)^{\frac{1}{w-1}}} \quad (3)$$

dimana:

$D(x, c)$ adalah jarak antara data x ke titik pusat kelompok c .

7. Hitung fungsi objektif pada persamaan 1.
8. Lakukan langkah 5, 6, 7 sehingga tidak ada perubahan signifikan pada fungsi objektif atau tahapan iterasi maksimal sudah terpenuhi.

B. Support Vector Regression (SVR)

SVR merupakan metode pengembangan dari Support Vector Machine untuk kasus prediksi dan regresi [12]. Seperti halnya SVM, SVR bertujuan untuk menciptakan *hyperplane* dan *support vector* untuk menentukan nilai prediksi. SVR memiliki performa yang handal dalam prediksi data *time-series*.

Diberikan data $\{(x_i, y_i)\}_{i=1}^N$, dimana x_i adalah vektor input dari data, y_i adalah nilai label dari data pelatihan, dan N adalah jumlah data pelatihan. Fungsi linear dari data tersebut dapat dilihat pada persamaan 4.

$$g(x_i) = (w \cdot x_i) + b \quad (4)$$

dimana :

$g(x_i)$ adalah nilai yang diprediksi, w adalah nilai bobot dan b adalah nilai bias.

Tujuan dari SVR adalah membentuk *hyperplane* dengan meminimalkan deviasi antara nilai yang diprediksi, $g(x_i)$, dan nilai yang sebenarnya. Hal ini dapat dilakukan dengan membentuk optimasi persamaan kuadrat pada persamaan 5.

Meminimalkan :

$$Z = \frac{1}{2} ||w^2||$$

dengan batasan :

$$y_i - w(x_i) - b \leq \varepsilon$$

$$w(x_i) - y_i + b \leq \varepsilon$$

Dalam kasus ini, diasumsikan bahwa semua titik berada pada rentang $(x) \pm \varepsilon$, kondisi demikian disebut sebagai permasalahan yang *feasible*. Sedangkan pada kenyataannya tidak selalu demikian, ada kemungkinan beberapa titik berada di luar area tersebut. Untuk mengatasi permasalahan ini, dapat ditambahkan *slack variable* yaitu ξ_i, ξ_i^* sehingga persamaan 5 dapat dirubah pada persamaan 6.

Meminimalkan :

$$Z = \frac{1}{2} ||w^2|| + C \sum_{i=1}^N (\xi_i + \xi_i^*) \quad (6)$$

dengan batasan:

$$y_i - (w \cdot x_i) - b \leq \varepsilon + \xi_i$$

$$(w \cdot x_i) + b - y_i \leq \varepsilon + \xi_i^*$$

$$\xi_i, \xi_i^* \geq 0$$

Nilai C merupakan nilai penalti yang diberikan jika hasil prediksi memiliki kesalahan melebihi ε . Persamaan optimasi kuadrat pada persamaan 6 dapat diselesaikan menggunakan Fungsi Lagrange sehingga didapatkan nilai bobot yang baru pada persamaan 7.

$$w = \sum_{i=1}^l (a_i - a_i^*) x_i \quad (7)$$

dimana :

a_i, a_i^* : Lagrange Multiplier

Dengan menggunakan nilai bobot pada persamaan 7, persamaan 4 dapat dirubah menjadi persamaan 8.

$$g(x_i) = (w \cdot x) + b = \sum_{i=1}^l (a_i - a_i^*) (x_i \cdot x) + b \quad (8)$$

dimana x_i merupakan titik paling informatif dalam SVR atau biasa disebut *Support Vector* (SV) yang mewakili keseluruhan informasi dari model SVR. Nilai b dapat dihitung menggunakan prinsip Karush Kuhn Tucker (KKT) sebagai berikut:

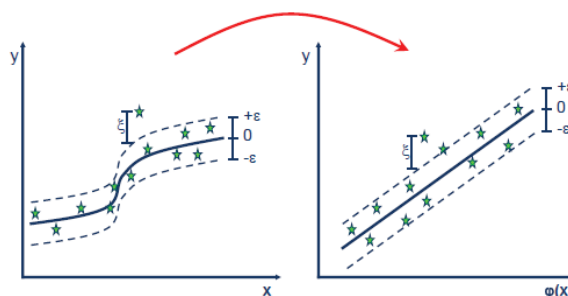
$$b = y_i - \langle w^* - x_i \rangle - \varepsilon, \quad 0 \leq a_i \leq C \quad (9)$$

$$b^* = y_i - \langle w^* - x_i \rangle + \varepsilon, \quad 0 \leq a_i^* \leq C \quad (10)$$

Untuk mengatasi permasalahan non-linear, fungsi kernel dapat diterapkan dalam SVR untuk memetakan data pelatihan ke dalam ruang fitur dengan dimensi yang lebih tinggi menggunakan persamaan 11 dan fungsi kernel pada persamaan 12. Ilustrasi pemetaan data pelatihan ke dalam ruang fitur dapat dilihat pada Gambar 1.

$$\varphi : R^n \rightarrow R^{n+h}; x \rightarrow \varphi(x) \quad (11)$$

$$K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j) \quad (12)$$



Gambar 1 Ilustrasi pemetaan data pelatihan ke dalam ruang fitur pada SVR

Salah satu fungsi kernel yang sering digunakan dalam SVR adalah kernel gaussian RBF. Kernel tersebut dapat memetakan data pelatihan ke dalam ruang fitur dan handal dalam menangani permasalahan non-linear. Fungsi kernel Gaussian RBF dapat dilihat pada persamaan 13.

$$K(x_i, x_j) = \exp\left(-\frac{1}{2\sigma^2} \|x_i - x_j\|^2\right) \quad (13)$$

Dalam penggunaan fungsi kernel linear dalam SVR, terdapat 3 parameter yang harus ditentukan, yaitu: nilai C sebagai nilai penalti, nilai ϵ sebagai nilai toleransi kesalahan, dan nilai σ yang menyatakan lebar dari fungsi gaussian RBF.

C. Particle Swarm Optimization (PSO)

PSO merupakan salah satu algoritma optimasi meta-heuristik dan merupakan bagian dari *swarm intelligence*. PSO memiliki konsep yang sederhana, efisien, dan cepat dalam mencapai solusi yang konvergen [13]. PSO didasarkan pada perilaku kawanan hewan untuk mencapai solusi yang optimal. Tindakan suatu individu dipengaruhi oleh memori posisi terbaik yang pernah dilalui dan informasi posisi terbaik dari seluruh individu dalam kawanan untuk mencapai solusi yang optimal.

Dalam PSO, *swarm* merupakan kumpulan dari beberapa individu yang menyatakan solusi yang potensial dalam ruang pencarian. Setiap solusi ini akan dievaluasi menggunakan fungsi *fitness*. Setiap individu memiliki memori terhadap posisi terbaik yang pernah dilalui (P_{best}). Solusi terbaik dari seluruh partikel dalam populasi merupakan solusi terbaik global (G_{best}). Dalam setiap iterasi dalam pencarian solusi, setiap individu akan memperbarui posisinya menggunakan informasi dari P_{best} dan G_{best} sehingga semua individu akan memberikan solusi yang konvergen dan dapat mencapai solusi yang optimal dalam ruang pencarian solusi.

Berikut ini merupakan tahapan dari algoritma PSO :

1. Tentukan jumlah partikel dalam populasi (*swarm*), posisi awal setiap partikel, jumlah iterasi maksimal I , fungsi *fitness*, konstanta inersia w , konstanta memori individu c_1 , konstanta pengaruh sosial c_2 , serta ambang batas fungsi *fitness*.
2. Hitung nilai fungsi objektif dari setiap partikel.
3. Bandingkan nilai fungsi evaluasi setiap partikel untuk menentukan P_{best} .
4. Tentukan nilai G_{best} dengan mendapatkan nilai terbaik dari P_{best} .
5. Perbarui kecepatan (persamaan 14) dan posisi (persamaan 15) setiap partikel dalam populasi.

$$v_{t+1} = wv_t + c_1r_1(P_{best} - x_t) + c_2r_2(G_{best} - x_t) \quad (14)$$

$$x_{t+1} = x_t + v_{t+1} \quad (15)$$

dimana :

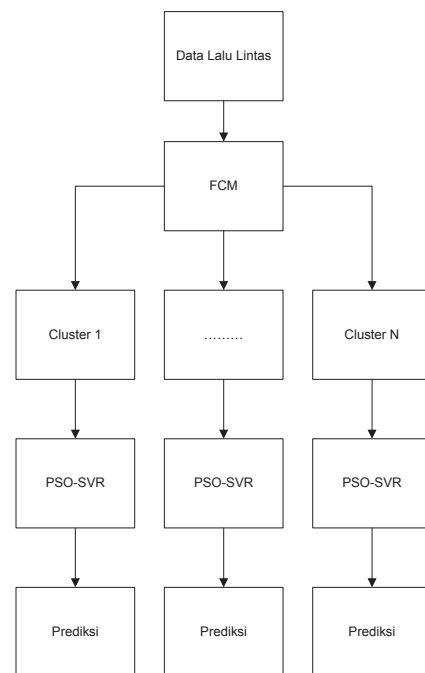
v_{t+1} = kecepatan partikel

r_1, r_2 = bilangan random [0,1]

6. Lakukan langkah 2, 3, 4, dan 5 hingga kondisi terminasi terpenuhi.

IV. DESAIN METODE

Metode yang kami usulkan merupakan metode hibrida pengelompokan dan prediksi data lalu lintas. Algoritma FCM digunakan untuk mengelompokkan data arus lalu lintas menjadi beberapa kelompok. Pada setiap kelompok yang terbentuk akan dibangun model prediksi SVR. Algoritma PSO digunakan sebagai metode optimasi pemilihan parameter yang akan digunakan pada SVR. Arsitektur dari metode yang diajukan dapat dilihat pada Gambar 2.



Gambar 2 Arsitektur metode FCM-PSO-SVR yang diusulkan

A. Pengelompokan Data Lalu Lintas

Data vektor masukan yang akan dikelompokkan dalam proses FCM adalah data arus lalu lintas yang akan digunakan untuk membangun model prediksi. Sebagai contoh, jika 3 data arus lalu lintas pada waktu sebelumnya digunakan untuk membangun model prediksi, maka vektor masukan yang digunakan dalam proses FCM adalah sebagai berikut:

Vektor Masukan = $\{ v(t-3), v(t-2), v(t-1) \}$, dimana $v(t)$ merupakan arus lalu lintas pada indeks waktu t .

TABEL IV
ILUSTRASI PROSES PENGELOMPOKAN DATA LALU LINTAS DENGAN FCM

| Waktu yang diprediksi | Vektor Masukan | | | Derajat Keanggotaan | | | kelompok data lalu lintas |
|-----------------------|----------------|--------|--------|---------------------|-----------------|-----------------|---------------------------|
| | v(t-1) | v(t-2) | v(t-3) | μ_1 | μ_2 | μ_3 | |
| 18.3 | 278 | 277 | 247.5 | 0.003883 | 0.022523 | 0.973594 | 3 |
| 18.45 | 277 | 247.5 | 214 | 0.021173 | 0.156023 | 0.822804 | 3 |
| 19 | 247.5 | 214 | 204.5 | 0.03587 | 0.460743 | 0.503386 | 3 |
| 19.15 | 214 | 204.5 | 178.5 | 0.026798 | 0.833991 | 0.139211 | 2 |
| 19.3 | 204.5 | 178.5 | 151 | 0.023604 | 0.922898 | 0.053498 | 2 |
| 19.45 | 178.5 | 151 | 134.5 | 0.052141 | 0.89558 | 0.052279 | 2 |
| 20 | 151 | 134.5 | 121.5 | 0.151076 | 0.772119 | 0.076805 | 2 |
| 20.15 | 134.5 | 121.5 | 103 | 0.318822 | 0.592067 | 0.089111 | 2 |
| 20.3 | 121.5 | 103 | 91.5 | 0.511184 | 0.409516 | 0.0793 | 1 |
| 20.45 | 103 | 91.5 | 87.5 | 0.672908 | 0.267253 | 0.059839 | 1 |
| 21 | 91.5 | 87.5 | 77 | 0.780917 | 0.175503 | 0.04358 | 1 |

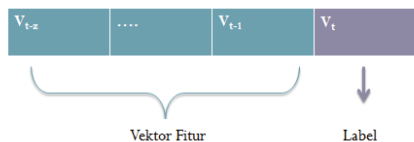
Nilai bobot w yang digunakan dalam FCM dalam penelitian ini adalah 2. Proses perhitungan jarak antara data i dan titik pusat kelompok j dalam algoritma FCM menggunakan jarak euclidean pada persamaan 16.

$$D(x_i, c_j) = \sqrt{(x_i - c_j)^2} \quad (16)$$

Dalam akhir proses FCM, kita akan mendapatkan data arus lalu lintas dalam setiap kelompok yang akan digunakan untuk membangun model prediksi SVR. Selain itu, kita juga akan mendapatkan titik pusat dari setiap kelompok yang dihasilkan yang akan digunakan dalam tahap pengujian. Ilustrasi proses pengelompokan data lalu lintas menggunakan FCM dapat dilihat pada Tabel 1.

B. Pembentukan Model SVR

Proses pembentukan model SVR dilakukan dalam setiap kelompok data lalu lintas yang dihasilkan pada tahap sebelumnya. Untuk pembentukan fitur pada SVR digunakan vektor fitur yang berisi data yang digunakan untuk memprediksi data ke t . Sebagai contoh, jika kita akan memprediksikan arus pada indeks t dengan 3 data sebelumnya, yaitu $\{v(t-3), v(t-2), v(t-1)\}$. Vektor tersebut merupakan vektor masukan yang digunakan dalam proses FCM. Representasi vektor fitur dan nilai target/label dapat dilihat pada Gambar 3.



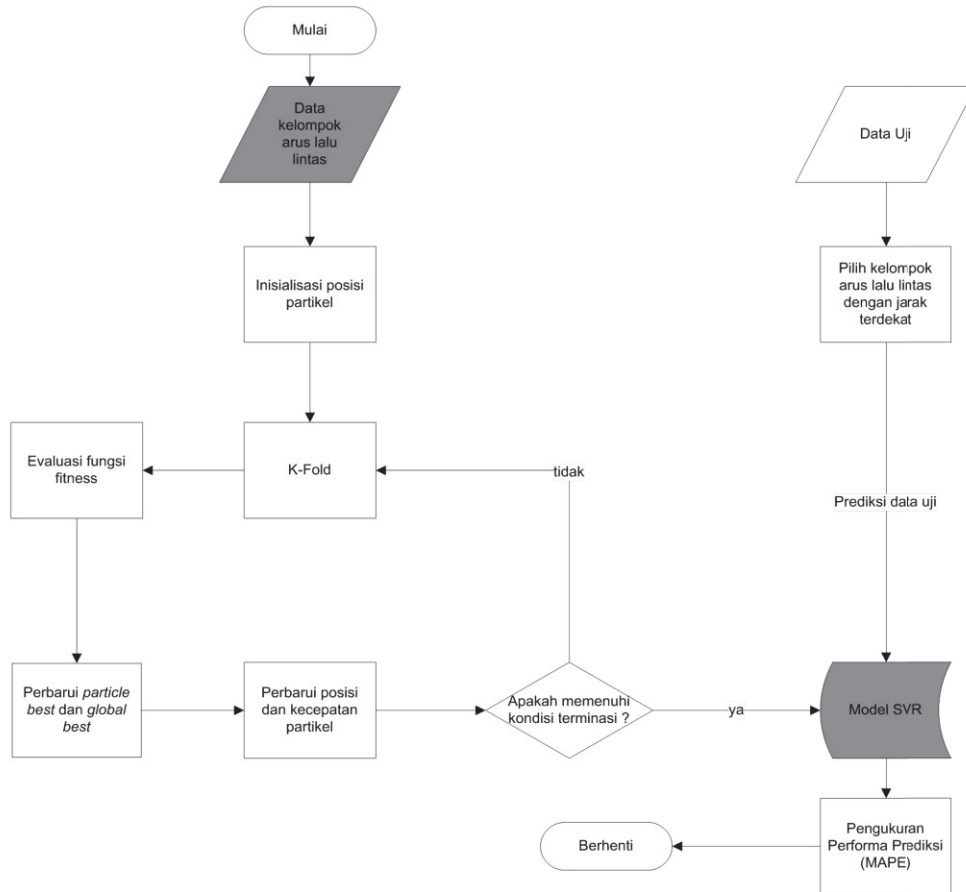
Gambar 3. Representasi vektor fitur dan label pada SVR

C. Model PSO-SVR

Algoritma PSO digunakan sebagai algoritma optimasi untuk pencarian nilai parameter yang sesuai untuk SVR dalam setiap kelompok data yang telah terbentuk. Dengan menggunakan fungsi kernel gaussian RBF, kita perlu menentukan nilai yang tepat untuk koefisien C sebagai nilai penalti, koefisien ε sebagai nilai toleransi kesalahan, dan koefisien σ yang menyatakan lebar dari fungsi gaussian RBF. Sehingga representasi setiap partikel dalam PSO sebagai berikut $\{C, \varepsilon, \sigma\}$. Metode PSO-SVR yang digunakan dapat dilihat pada Gambar 4. Bagian berwarna abu-abu merupakan kontribusi dari penelitian ini.

1) *Algoritma PSO-SVR*: Berikut ini merupakan algoritma PSO dalam penentuan parameter SVR:

- Inisialisasi parameter PSO untuk konstanta inersia w , konstanta pengaruh memori individu c_1 , konstanta pengaruh sosial c_2 , jumlah iterasi maksimal I , ambang batas FT untuk solusi yang dihasilkan, jumlah partikel yang digunakan, dan posisi setiap partikel dalam populasi.
- Hitung nilai dari fungsi *fitness* setiap partikel dalam populasi.
- Tentukan nilai P_{best} untuk setiap partikel dan nilai G_{best} untuk menentukan solusi terbaik dari kawanan dari seluruh partikel dalam kawanan.
- Perbarui kecepatan dan posisi setiap partikel untuk iterasi selanjutnya menggunakan persamaan 14 dan 15.



Gambar 4 Desain Metode PSO-SVR

e) Lakukan langkah b, c, d hingga jumlah iterasi maksimal I tercapai atau solusi yang dihasilkan di bawah ambang batas FT.

2) *Fungsi Fitness dalam PSO-SVR*: Metode K-Fold *cross validation* dan nilai Mean Absolute Percentage Error (MAPE) digunakan sebagai fungsi evaluasi dari solusi yang dihasilkan oleh setiap partikel dalam PSO. Berikut merupakan tahapan metode K-Fold dan fungsi evaluasi dalam PSO-SVR :

a) Bagi data pelatihan menjadi K bagian secara acak dengan ukuran yang sama, D_1, D_2, \dots, D_K .

b) Untuk $k= 1, 2, \dots, K$

- Gunakan data D_K sebagai data validasi dan data lainnya sebagai data pelatihan.
- Latih model SVR dengan menggunakan parameter yang diberikan oleh PSO.
- Hitung nilai *Mean Absolute Percentage Error* (MAPE) pada persamaan 17 untuk mengukur performa prediksi SVR dengan parameter yang diberikan oleh PSO.

$$MAPE = \frac{1}{n} \times \sum_{i=1}^n \left| \frac{x_i - x'_i}{x_i} \right| \quad (17)$$

dimana :

n adalah banyak data yang diprediksi

x_i adalah nilai tingkat arus lalu lintas aktual

x'_i adalah nilai tingkat arus lalu lintas yang diprediksi

c) Hitung nilai rata-rata MAPE sebagai nilai dari fungsi *fitness* untuk setiap partikel PSO.

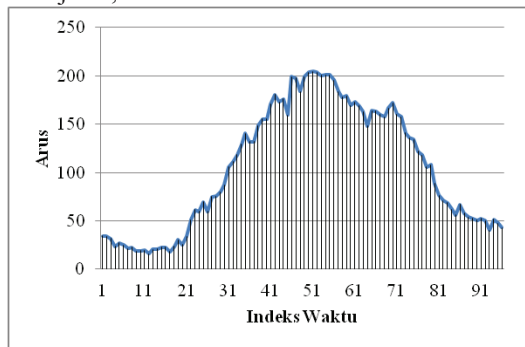
3) *Tahap Pengujian dan Pengukuran Performa Model FCM-PSO-SVR*: Parameter terbaik yang diberikan oleh PSO digunakan untuk melatih model SVR dalam setiap kelompok data yang terbentuk. Proses pengujian dilakukan dengan mencari kelompok yang terdekat dengan data uji. Model SVR dari kelompok yang terdekat akan digunakan untuk memprediksi data uji. Proses perhitungan jarak antara data uji i dengan titik pusat kelompok data j dapat dihitung menggunakan jarak euclidean dalam persamaan 16. Nilai MAPE dalam persamaan 17 digunakan untuk mengukur performa dari metode yang diajukan.

V. DATASET

Dataset yang digunakan dalam penelitian ini diambil dari <http://data.gov.uk/dataset/dft-eng-srn-routes-journey-times>. Dataset tersebut merupakan data lalu lintas yang dihimpun oleh *Traffic Highways Agency* pada berbagai lokasi jalan

(motorway) di negara Inggris. Dataset berisi data lalu lintas dalam kurun waktu 1 bulan dengan interval waktu pengamatan 15 menit sehingga terdapat 96 data pengamatan dalam 1 hari. Data tersebut memiliki beberapa atribut, yaitu: arus lalu lintas, rata-rata kecepatan, dan waktu perjalanan. Dalam penelitian ini atribut yang digunakan adalah arus lalu lintas. Contoh plot data arus lalu lintas pada lokasi 'AL100', hari Senin, 3 November 2014 dapat dilihat pada Gambar 5.

Dalam penelitian ini, kami hanya menggunakan data pada hari kerja (Senin sampai Jumat). Hal ini dikarenakan kondisi arus lalu lintas yang sangat berbeda antara hari kerja dan hari libur. Data yang digunakan dalam penelitian adalah data lalu lintas dalam kondisi normal tanpa ada kejadian istimewa seperti cuaca ekstrim, terjadinya kecelakaan, perbaikan jalan, dsb.



Gambar 5 Plot data Arus Lalu Lintas pada lokasi 'AL100', 3 November 2014

VI. UJICOBA DAN PEMBAHASAN

Implementasi dari metode yang diusulkan dibangun pada lingkungan pengembangan Matlab R2012A. Kakas bantu LibSVM [14] digunakan untuk membentuk model prediksi SVR dengan fungsi kernel gaussian RBF. Parameter PSO yang digunakan dalam PSO-SVR dapat dilihat pada Tabel 2. Dalam data 1 bulan pengamatan, data arus lalu lintas pada minggu pertama hingga minggu ketiga digunakan sebagai data pelatihan. Sedangkan data minggu keempat digunakan sebagai data uji.

TABEL V
PARAMETER PSO DALAM PSO-SVR

| w | c ₁ | c ₂ | iterasi maksimal | jumlah partikel |
|---|----------------|----------------|------------------|-----------------|
| 1 | 2 | 2 | 15 | 45 |

Ujicoba pertama bertujuan untuk membandingkan model prediksi PSO-SVR dan FCM-PSO-SVR yang dibangun dengan menggunakan data seluruh hari kerja. Data yang digunakan adalah data lalu lintas pada lokasi 'AL100'. Proses pengelompokan data dengan FCM dan pembentukan vektor fitur pada SVR menggunakan fitur data 45 menit $\{v(t-3), v(t-2), v(t-1)\}$ dan data 60 menit $\{v(t-4), v(t-3), v(t-2), v(t-1)\}$ untuk memprediksikan arus lalu lintas pada indeks waktu t . Jumlah kelompok yang dibentuk menggunakan FCM adalah 3, 4, dan 5 kelompok. Dapat dilihat pada Tabel 3 FCM-PSO-SVR memberikan performa yang lebih baik daripada metode PSO-SVR untuk seluruh hari yang diuji. Tabel 4 menunjukkan metode FCM-PSO-SVR juga memberikan hasil yang lebih baik dibandingkan PSO-SVR dengan menggunakan data 60 menit sebelumnya.

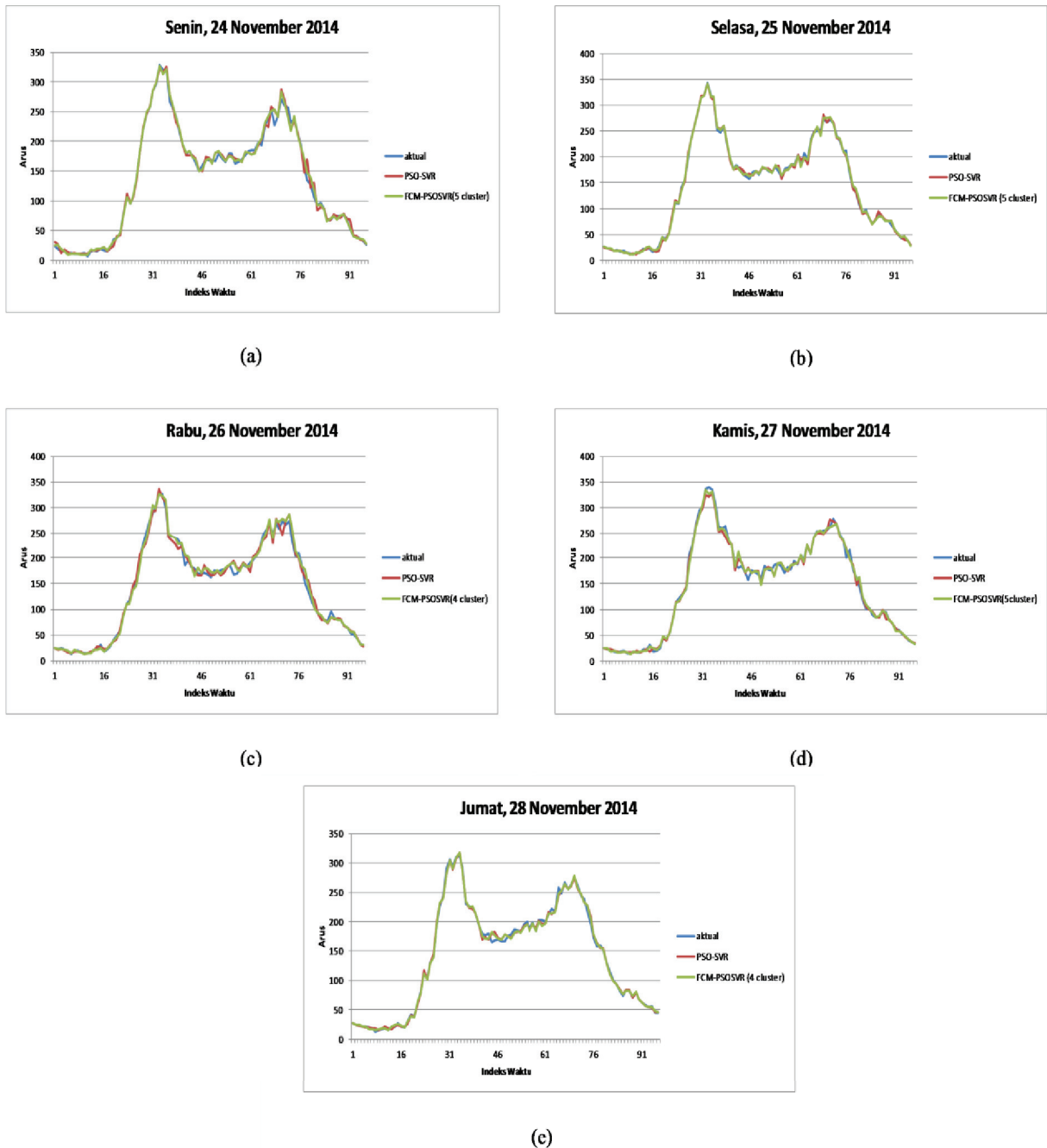
TABEL VI
NILAI MAPE(%) PADA UJICOBA PERTAMA UNTUK PREDIKSI DENGAN DATA 45 MENIT SEBELUMNYA PADA LOKASI 'AL100'

| Hari | PSO-SVR | FCM-PSO-SVR | | |
|-----------|---------|-------------|--------------|--------------|
| | | 3 cluster | 4 cluster | 5 cluster |
| Senin | 9.059 | 7.612 | 7.599 | 8.343 |
| Selasa | 5.847 | 5.841 | 5.575 | 5.349 |
| Rabu | 6.983 | 6.882 | 6.841 | 6.698 |
| Kamis | 8.309 | 7.705 | 7.881 | 7.719 |
| Jumat | 4.906 | 4.769 | 4.844 | 4.357 |
| Rata-rata | 7.02 | 6.562 | 6.548 | 6.493 |

TABEL VII
NILAI MAPE(%) PADA UJICOBA PERTAMA UNTUK PREDIKSI DENGAN DATA 60 MENIT SEBELUMNYA PADA LOKASI 'AL100'

| Hari | PSO-SVR | FCM-PSO-SVR | | |
|-----------|---------|-------------|--------------|--------------|
| | | 3 cluster | 4 cluster | 5 cluster |
| Senin | 9.799 | 7.874 | 7.069 | 7.883 |
| Selasa | 5.018 | 5.066 | 4.617 | 4.664 |
| Rabu | 6.376 | 5.439 | 5.263 | 6.085 |
| Kamis | 7.131 | 6.394 | 5.198 | 5.388 |
| Jumat | 4.972 | 4.596 | 4.528 | 4.221 |
| Rata-rata | 6.659 | 5.874 | 5.335 | 5.648 |

hari yang diuji dapat dilihat pada Gambar 6. Metode FCM-



Gambar 6 Perbandingan metode prediksi pada ujicoba kedua pada lokasi 'AL100'

Ujicoba kedua bertujuan untuk membandingkan metode PSO-SVR (tanpa proses *clustering*) dan FCM-PSO-SVR yang dibangun dengan menggunakan data di hari yang sama. Jumlah kelompok yang digunakan adalah 3, 4, dan 5 kelompok. Jumlah data yang digunakan untuk vektor masukan dalam FCM dan pembentukan model SVR adalah data 60 menit $\{ v(t-4), v(t-3), v(t-2), v(t-1) \}$ untuk memprediksi nilai arus pada indeks t . Plot perbandingan hasil prediksi PSO-SVR dan FCM-PSO-SVR untuk semua

PSO-SVR memberikan performa yang lebih akurat dibandingkan metode PSO-SVR dikarenakan dalam metode FCM-PSO-SVR memiliki beberapa metode prediksi sesuai dengan jumlah kelompok yang digunakan.

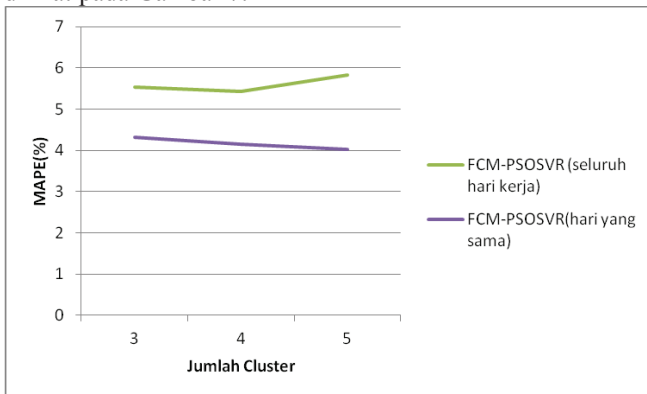
Sebagai contoh dengan membentuk 4 kelompok, maka akan terdapat 4 model prediksi yang dibangun untuk hari tersebut. Hasil prediksi menjadi lebih akurat karena untuk melakukan prediksi menggunakan model yang lebih sesuai dengan data uji. Dengan menggunakan jumlah cluster yang

berbeda performa metode prediksi PSO-SVR dapat meningkat seperti yang terlihat pada Tabel 5.

TABEL VIII
NILAI MAPE(%) PADA UJICOBA KEDUA DENGAN DATA 60 MENIT SEBELUMNYA PADA LOKASI 'AL100'

| Hari | PSO-SVR | FCM-PSO-SVR | | |
|--------|---------|-------------|--------------|--------------|
| | | 3 cluster | 4 cluster | 5 cluster |
| Senin | 6.224 | 5.755 | 5.889 | 5.177 |
| Selasa | 3.620 | 3.141 | 3.604 | 3.066 |
| Rabu | 5.984 | 4.691 | 4.124 | 4.789 |
| Kamis | 5.356 | 4.610 | 4.207 | 3.891 |
| Jumat | 3.504 | 3.373 | 2.979 | 3.201 |

Performa metode prediksi PSO-SVR pada ujicoba kedua (dengan menggunakan hari yang sama) memiliki performa yang lebih baik daripada PSO-SVR pada ujicoba pertama (dengan menggunakan seluruh hari kerja). Hal tersebut dikarenakan pada hari yang sama memiliki karakteristik yang serupa. Begitu juga dengan metode FCM-PSO-SVR yang dibangun menggunakan hari yang sama (ujicoba kedua) juga memiliki performa lebih baik dibandingkan metode FCM-PSO-SVR yang dibangun menggunakan keseluruhan hari kerja (ujicoba pertama). Perbandingan FCM-PSO-SVR pada ujicoba pertama dan kedua dapat dilihat pada Gambar 7.



Gambar 7 Perbandingan hasil prediksi FCM-PSO-SVR pada lokasi 'AL100'

Selanjutnya, kami menguji metode FCM-PSO-SVR yang dibangun dengan menggunakan hari yang sama untuk 4 lokasi yang berbeda, yaitu: 'AL1001', 'AL1004', 'AL1007', dan 'AL1073'. Vektor masukan dalam FCM dan vektor fitur pada SVR menggunakan data arus 60 menit sebelumnya. Perbandingan hasil prediksi metode FCM-PSO-SVR dan PSO-SVR untuk setiap lokasi yang diuji dapat dilihat pada Tabel 6, Tabel 7, Tabel 8, dan Tabel 9. Dapat diketahui dari hasil ujicoba tersebut bahwa metode pengelompokan FCM dapat meningkatkan performa dari metode prediksi PSO-SVR. Pemilihan jumlah kelompok menjadi hal yang penting untuk menghasilkan metode prediksi yang memberikan performa terbaik.

TABEL IX
NILAI MAPE(%) DENGAN DATA 60 MENIT SEBELUMNYA DENGAN MENGGUNAKAN MODEL PADA HARI YANG SAMA PADA LOKASI 'AL1001'

| Hari | PSO-SVR | FCM-PSO-SVR | | |
|--------|---------|-------------|--------------|--------------|
| | | 3 cluster | 4 cluster | 5 cluster |
| Senin | 6.812 | 5.944 | 5.516 | 6.421 |
| Selasa | 3.709 | 3.054 | 3.181 | 3.001 |
| Rabu | 6.103 | 4.950 | 5.597 | 4.150 |
| Kamis | 5.448 | 4.301 | 4.656 | 3.951 |
| Jumat | 4.510 | 3.361 | 3.172 | 3.315 |

TABEL X
NILAI MAPE(%) DENGAN DATA 60 MENIT SEBELUMNYA DENGAN MENGGUNAKAN MODEL PADA HARI YANG SAMA PADA LOKASI 'AL1004'

| Hari | PSO-SVR | FCM-PSO-SVR | | |
|--------|---------|--------------|--------------|--------------|
| | | 3 cluster | 4 cluster | 5 cluster |
| Senin | 2.647 | 2.936 | 3.854 | 2.438 |
| Selasa | 1.843 | 1.771 | 2.779 | 1.936 |
| Rabu | 2.875 | 2.695 | 2.317 | 4.068 |
| Kamis | 1.747 | 1.594 | 1.388 | 1.743 |
| Jumat | 2.695 | 2.414 | 2.216 | 2.553 |

TABEL XI
NILAI MAPE(%) DENGAN DATA 60 MENIT SEBELUMNYA DENGAN MENGGUNAKAN MODEL PADA HARI YANG SAMA PADA LOKASI 'AL1007'

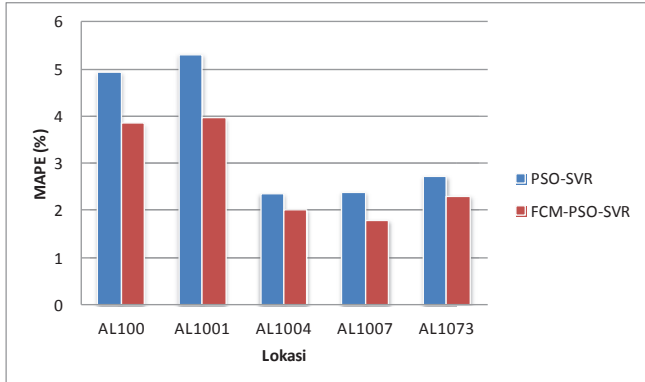
| Hari | PSO-SVR | FCM-PSO-SVR | | |
|--------|---------|-------------|--------------|--------------|
| | | 3 cluster | 4 cluster | 5 cluster |
| Senin | 2.708 | 3.191 | 2.563 | 2.656 |
| Selasa | 2.544 | 1.523 | 1.602 | 1.302 |
| Rabu | 2.952 | 2.442 | 2.234 | 2.275 |
| Kamis | 2.108 | 1.966 | 1.431 | 1.416 |
| Jumat | 1.641 | 1.725 | 1.452 | 1.658 |

TABEL XII
NILAI MAPE(%) DENGAN DATA 60 MENIT SEBELUMNYA DENGAN MENGGUNAKAN MODEL PADA HARI YANG SAMA PADA LOKASI 'AL1073'

| Hari | PSO-SVR | FCM-PSO-SVR | | |
|--------|---------|--------------|--------------|--------------|
| | | 3 cluster | 4 cluster | 5 cluster |
| Senin | 3.931 | 3.627 | 2.704 | 3.776 |
| Selasa | 1.780 | 2.304 | 2.059 | 1.617 |
| Rabu | 2.757 | 2.831 | 2.642 | 3.237 |
| Kamis | 2.575 | 2.373 | 2.644 | 2.376 |
| Jumat | 2.587 | 2.201 | 2.804 | 2.352 |

Perbandingan performa metode prediksi PSO-SVR dan

FCM-PSO-SVR pada 5 lokasi yang dicoba dapat dilihat pada Gambar 8. Metode FCM-PSO-SVR mampu memberikan performa lebih baik daripada PSO-SVR untuk semua lokasi yang diuji. Selanjutnya, FCM-PSO-SVR dengan jumlah kelompok terbaik dapat memberikan performa prediksi dengan kesalahan di bawah 4% pada semua lokasi yang diuji.



Gambar 8 Perbandingan performa metode prediksi PSO-SVR dan FCM-PSO-SVR untuk semua lokasi yang diuji.

VII. KESIMPULAN

Dalam penelitian ini kami telah mengusulkan metode hibrida pengelompokan dan prediksi arus lalu lintas dengan menggunakan FCM dan PSO-SVR. Hasil ujicoba menunjukkan bahwa metode pengelompokan dengan menggunakan FCM dapat meningkatkan performa dari metode PSO-SVR. Selanjutnya metode FCM-PSO-SVR yang dibangun pada hari yang sama memiliki performa yang lebih baik daripada metode PSO-SVR yang dibangun menggunakan seluruh hari kerja dan PSO-SVR yang dibangun pada hari yang sama. Metode FCM-PSO-SVR mampu memberikan performa prediksi dengan kesalahan di bawah 4% untuk semua lokasi yang diuji.

Dalam penelitian ini, kami menggunakan data lalu lintas dalam kondisi normal. Penelitian selanjutnya dapat berfokus pada prediksi data lalu lintas dengan memperhatikan kondisi istimewa seperti perubahan cuaca ekstrim, terjadinya kecelakaan, perawatan jalan. dsb.

DAFTAR PUSTAKA

- [1] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," *Transp. Res. Part C Emerg. Technol.*, vol. 43, pp. 3–19, 2014.
- [2] M. Lippi, M. Bertini, and P. Frasconi, "Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 871–882, 2013.
- [3] W. C. Hong, Y. Dong, F. Zheng, and C. Y. Lai, "Forecasting urban traffic flow by SVR with continuous ACO," *Appl. Math. Model.*, vol. 35, no. 3, pp. 1282–1291, 2011.
- [4] W.-C. Hong, Y. Dong, F. Zheng, and S. Y. Wei, "Hybrid evolutionary algorithms in a SVR traffic flow forecasting model," *Appl. Math. Comput.*, vol. 217, no. 15, pp. 6733–6747, 2011.
- [5] Y. Deng, J. Xiang, and Z. Ou, "SVR with hybrid chaotic genetic algorithm for short-term traffic flow forecasting," in *Natural Computation (ICNC), 2012 Eighth International Conference on*, 2012, no. Icn, pp. 708–712.
- [6] R. X. Fang, W. H. Yao, and X. D. Zhang, "Short-Term Traffic Flow Forecasting Based on SVR with Improved Artificial Fish Swarm Algorithm," *Appl. Mech. Mater.*, vol. 734, pp. 508–514, 2015.
- [7] Y. Kara, M. Acar Boyacioglu, and Ö. K. Baykan, "Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange," *Expert Syst. Appl.*, vol. 38, no. 5, pp. 5311–5319, 2011.
- [8] Z. Dong, D. Yang, T. Reindl, and W. M. Walsh, "A novel hybrid approach based on self-organizing maps, support vector regression and particle swarm optimization to forecast solar irradiance," *Energy*, vol. 82, no. 0, pp. 570–577, 2015.
- [9] C.-J. Lu and C.-C. Chang, "A Hybrid Sales Forecasting Scheme by Combining Independent Component Analysis with K-Means Clustering and Support Vector Regression," *Sci. World J.*, vol. 2014, no. July, pp. 1–8, 2014.
- [10] J. Kianfar and P. Edara, "A Data Mining Approach to Creating Fundamental Traffic Flow Diagram," in *Procedia - Social and Behavioral Sciences*, 2013, vol. 104, pp. 430–439.
- [11] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Comput. Geosci.*, vol. 10, no. 2–3, pp. 191–203, 1984.
- [12] H. Drucker and C. Burges, "Support vector regression machines," *Adv. Neural Inf. Process. Syst.*, vol. 9, pp. 155–161, 1997.
- [13] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Neural Networks, 1995. Proceedings., IEEE Int. Conf.*, vol. 4, pp. 1942–1948 vol.4, 1995.
- [14] C. Chih-Chung and L. Chih-Jen, "LIBSVM: A Library for support vector machines," *ACM Trans. Intell. Transp. Syst.*, vol. 2, no. 3, pp. 27:1 – 27:27, 2011.