Komparasi Metode *Machine Learning* dan *Non Machine Learning* untuk Estimasi Usaha Perangkat Lunak

Ega Kartika Adhitya, Romi Satria Wahono dan Hendro Subagyo
Fakultas Ilmu Komputer, Universitas Dian Nuwantoro
egaegi25gmail.com, romi@romisatriawahono.net, hendro.subagyo@gmail.com

Abstrak: Estimasi usaha perangkat lunak (EUPL) adalah proses yang sangat penting dalam mewujudkan kesuksesan pelaksanaan suatu proyek perangkat lunak. Ada banyak metode EUPL sehingga diperlukan pemilihan yang sesuai proyek yang akan dikembangkan agar menghasilkan estimasi yang akurat. Dataset yang sering digunakan untuk penelitian EUPL adalah dataset Albercht dan Desherhanis, namun dataset tersebut memiliki atribut yang tidak relevan yang dapat mengurangi akurasi estimasi. Penelitian ini mengkomparasi metode machine learning (ML) dan non machine learning (non ML) untuk mengetahui metode mana yang paling baik untuk EUPL serta pengunaan seleksi atribut *forward selection* (FS) dan *backward elimination* (BE) untuk menyelesaikan masalah atribut yang tidak relevan. ML terdiri dari algoritma *k-nearnest neigbhors* (kNN), *neural networks* (NN) dan *support vector machine* (SVM) sedangkan non ML terdiri dari metode *function point* (FP) dan *use case point* (UCP). Dari hasil eksperimen, kNN adalah metode terbaik pada ML dengan nilai RMSE 6.2 dan 9.4 untuk dataset Albercth dan Desherhanis. Sedangkan pada non ML, FP yang paling baik dengan nilai RMSE 4.6 dan 21.3 untuk dataset Albercth dan Desherhanis. Sementara hasil eksperimen berikutnya, kNN diintegrasikan dengan FS mendapat nilai RMSE 2.5 dan 6.4 untuk dataset Albercth dan Desherhanis lebih baik dari integrasi kNN dengan BE dengan nilai RMSE 7.5 baik untuk dataset Albercth maupun Desherhanis. Dari eksperimen, dapat disimpulkan bahwa metode ML lebih baik dari pada non ML dalam estimasi usaha perangkat lunak dengan kNN sebagai estimatornya. Integrasi kNN dengan FS juga terbukti mampu meningkatkan estimasi lebih baik daripada kNN standar maupun kNN dengan BE.

Kata kunci: estimasi usaha perangkat lunak, metode *machine learning*, metode *non-machine learning*, seleksi atribut

1 PENDAHULUAN

Perangkat lunak merupakan abstraksi fisik yang memungkinkan kita untuk berbicara dengan mesin perangkat keras (Dennis, 2012). Tanpa adanya perangkat lunak, maka perangkat keras yang telah diciptakan tidak akan dapat berguna atau berfungsi dengan optimal. Perangkat lunak tidak hanya berupa program komputer, tetapi juga berisi dokumentasi yang terkait dan data konfigurasi yang diperlukan untuk membuat sebuah program dapat beroperasi dengan benar.

Estimasi usaha perangkat lunak dibutuhkan karena dalam pengembangan perangkat lunak dibatasi oleh biaya dan jadwal yang telah ditentukan (Ian Sommerville, 2011) Salah satu kegiatan manajemen proyek perangkat lunak adalah kegiatan estimasi usaha perangkat lunak. Estimasi usaha perangkat lunak merupakan kegiatan memperkirakan berapa banyak sumber daya yang dibutuhkan untuk menyelesaikan sebuah rencana proyek (Ian Sommerville, 2011). Melakukan estimasi atau memperkirakan berapa jumlah tenaga dan waktu yang dibutuhkan untuk membuat sebuah sistem tidak mudah dilakukan karena diperlukan pemahaman yang benar tentang sebuah pekerjaan.

Kesuksesan proyek pengembangan dipengaruhi oleh banyak hal, diantaranya dukungan eksekutif, keterlibatan pengguna dalam proyek, pengalaman manager proyek, tujuan bisnis yang jelas, infrastruktur perangkat lunak dan penggunaan metodologi pengembangan yang formal. Faktor lainnya merupakan faktor yang berkaitan dengan *timing* dan *scope* proyek, diantaranya *scope* yang minimal dan estimasi yang handal (Albrecht & Gaffney, 1983). Estimasi sumber daya, biaya, dan jadwal untuk pengembangan perangkat lunak membutuhkan pengalaman, akses ke informasi sejarah proyek yang baik, dan keberanian untuk melakukan prediksi kuantitatif sedangkan informasi kualitatif adalah semua fakta yang ada. Estimasi membawa risiko yang melekat dan risiko ini dapat menyebabkan ketidak pastian dalam proyek.

Estimasi usaha pengembangan perangkat lunak adalah proses melakukan estimasi usaha yang diperlukan untuk pengembangan perangkat lunak (Nunes, Constantine, & Kazman, 2011). Perusahaan-perusahaan pengembang perangkat lunak banyak yang dihadapkan dengan masalah estimasi biaya dan waktu yang diperlukan untuk proyek pengembangan perangkat lunak. Estimasi jadwal proyek sulit dilakukan karena banyak hal, misalnya perangkat lunak yang dikembangkan kemungkinan harus berjalan di lingkungan sistem yang sama sekali berbeda, menggunakan pengembangan dengan teknologi baru atau kompetensi orang-orang yang terlibat dalam proyek yang mungkin tidak diketahui. Ada begitu banyak ketidakpastian sehingga sulit untuk memperkirakan biaya dan waktu pengembangan secara akurat selama awal tahapan proyek. (Albrecht & Gaffney, 1983)

Dalam perkembangannya metode-metode untuk estimasi usaha pengembangan perangkat lunak, metode-metode tersebut di kelompokan menjadi metode non machine learning dan machine learning (Shepperd & MacDonell, 2012), banyak metode estimasi telah diusulkan untuk estimasi usaha pengembangan perangkat lunak. Beberapa metode non-machine learning yang telah diusulkan dalam estimasi usaha perangkat lunak diantaranya *slim* (Adriano L I Oliveira, Braga, Lima, & Cornélio, 2010), cocomo (El-Sebakhy, 2011), *expert judgement* (Nassif, Capretz, & Hill, 1993), *function point* (FP) (Choy, Tang, & Tong, 2011)dan *use case point* (UCP) (Boehm & Papaccio, 1988).

Metode-metode non machine learning dalam estimasi usaha pengembangan perangkat lunak yang disebutkan sebelumnya merupakan metode konvensional yang memiliki tingkat keakuratan yang relatif rendah (Nunes, Constantine, & Kazman, 2011). Selain metode-metode non-machine learning di atas, belakangan ini dikembangkan estimasi dengan metode machine learning. Beberapa metode machine learning yang telah digunakan untuk estimasi usaha pengembangan perangkat lunak diantaranya adalah *case-based reasoning* (CBR) learning *k-nearnest neighbor* (kNN) (Ian Sommerville,

Copyright © 2015 IlmuKomputer.Com

2011),*neural networks* (NN), *support vector machine* (SVM) (Wen, Li, Lin, Hu, & Huang, 2012),*liner regression* (LR) (W. Wang & Zhou, 2012).

Namun dalam metode machine learning juga terdapat kekurangan pada tingkat keakuratan yang relatif rendah (Ian Sommerville, 2011) dataset yang digunakan diantaranya Cocomo, Desharnais, Maxwell dan Albercht, umumnya, estimasi usaha sangat sulit dilakukan dalam proyek perangkat lunak karena proyek perangkat lunak bersifat dinamis dan kesulitan menemukan proyek yang sangat mirip dengan proyek tersebut sebelumnya. Selain itu, sulit untuk menentukan metode estimasi yang cocok untuk proyek tersebut. Dataset *software effort* banyak memiliki atribut yang beragam. Misalnya dataset Albrecht memiliki 8 atribut, dataset deshernanis memiliki 8 atribut. Kinerja yang lebih baik dapat dicapai dengan menghilangkan beberapa atribut, yang dengan ini dapat menghilangkan atribut yang tidak relevan (Nassif, Capretz, & Ho, 2012). Seleksi atribut dapat digunakan untuk menentukan *subset of features* yang optimal, dimana hal ini dapat meningkatkan akurasi estimasi (S. Wang, Li, Song, Wei, & Li, 2011)

Seleksi atribut merupakan bagian penting untuk mengoptimalkan kinerja dari *classifier* (Mehmood, S. Palli, & Khan, 2014). Seleksi atribut dapat didasarkan pada pengurangan ruang fitur yang besar, misalnya dengan mengeliminasi atribut yang kurang relevan (Danger, Segura-Bedmar, Martínez, & Rosso, 2010). Penggunaan algoritma seleksi atribut yang tepat dapat meningkatkan *accuracy* Algoritma seleksi atribut dapat dibedakan menjadi dua tipe, yaitu *filter* dan *wrapper* (Danger et al., 2010). Contoh dari tipe *filter* adalah *information gain* (IG), *chi-square*, dan *log likelihood ratio*. Contoh dari tipe *wrapper* adalah *forward selection* dan *backward elimination* (Nassif et al., 2012). Hasil *precision* dari tipe *wrapper* lebih tinggi daripada tipe *filter*, tetapi hasil ini tercapai dengan tingkat kompleksitas yang besar. Masalah kompleksitas yang tinggi juga dapat menimbulkan masalah (W. Wang & Zhou, 2012).

Dari semua hasil penelitian yang sudah dilakukan belum ditemukan model yang paling tepat untuk estimasi usaha perangkat lunak. Maka dari itu akan dilakukan komparasi terhadap beberapa metode machine learning (kNN, SVM dan NN), dan metode non-machine learning (FP dan UCP), komparasi terhadap beberapa seleksi atribut (*forward selection*, *backward elimination*) dan melakukan integrasi dari hasil komparasi metode machine learning, non-machine learning dan seleksi atribut yang terbaik pada metode estimasi usaha perangkat lunak, sehingga di dapatkan model yang baik untuk estimasi usaha perangkat lunak.

2 PENELITIAN TERKAIT

Nassif (Nassif et al., 2012), melakukan penelitian tentang perbandingan akurasi yang dihasilkan oleh algoritma *neural network* dan *linear regression,* dan *linear regression* secara bersama-sama akan digunakan dengan *algorithmic* model, yaitu *use case point*, dataset yang digunakan Albercth dengan hasil MMER :LR: 39.2 dan NN: 40

Kocaguneli (Kocaguneli & Menzies, 2013) melakukan penelitian tentang perbandingan akurasi yang bertujuan untuk mengkarakterisasi isi penting dari data *software effort estimation* (SEE), yaitu, sedikitnya jumlah fitur dan contoh yang diperlukan untuk menangkap informasi dalam data yang ada. Jika isi penting sangat kecil, maka informasi yang terkandung harus sangat singkat dan nilai tambah skema pembelajaran yang kompleks harus minimal, dataset yang digunakan cocomo dengan hasil RMSE kNN : 41.42

Adriano (Adriano L.I. Oliveira, Braga, Lima, & Cornélio, 2010), melakukan penelitian tentang analisis industri perangkat lunak, manajer proyek biasanya mengandalkan pengalaman mereka sebelumnya untuk memperkirakan jumlah atau waktu yang dibutuhkan untuk setiap proyek perangkat lunak. Keakuratan perkiraan tersebut merupakan faktor kunci untuk aplikasi yang efisien bagi sumber daya manusia. Metode yang digunaknan yaitu teknik pembelajaran mesin seperti fungsi radial basis (RBF) jaringan saraf, *multi-layer perceptron* (MLP) jaringan saraf, regresi dukungan vektor (SVR), mengantongi prediktor dan pohon regresi berbasis, untuk memperkirakan perangkat lunak. Beberapa karya telah menunjukkan bahwa tingkat akurasi estimasi usaha software sangat tergantung pada nilai-nilai parameter metode ini. Selain itu, telah ditunjukkan bahwa pemilihan fitur masukan juga mungkin memiliki pengaruh penting pada akurasi estimasi, dataset yang digunakan desherhanis dengan hasil RMSE SVM 83.33

Nuno (Nunes et al., 2011a), telah mengusulkan beberapa metode pengukuran dengan ukuran function al dan biaya model estimasi, terutama pada poin fungsi analisis (FPA) dan Cocomo. Keduanya menganggap bahwa pengembang dapat memperoleh pengukuran ukuran dan perkiraan dari sejarah data proyek dan karakteristik proyek saat ini. Dengan orientasi objek, menggunakan kasus yang muncul sebagai sebuah teknik yang dominan untuk persyaratan penataan. Teknik ini telah dikomparasikan ke dalam *unified modeling language* (UML) dan *unified process* dan menjadi standar sesungguhnya untuk persyaratan peragaan, dataset yang digunakan privat dengan hasil UCP 85.33

Felfernig (Dubois, Rasovska, & De Guio, 2009), melakukan fungsi point analysis (FPA) untuk berorientasi objek, pengembangan estimasi usaha perangkat lunak. Dalam makalah ini menggunakan fungsi point analysis (FPA) (Papatheocharous & Andreou, 2009) dapat diterapkan untuk estimasi upaya pengembangan sistem konfigurasi berbasis pengetahuan. FPA didasarkan pada user dan berpusat tampilan pada perangkat lunak dan *platform-independen*. Metode tersebut pertama kali diusulkan oleh Alberth (Albrecht & Gaffney, 1983) dengan tujuan untuk memberikan upaya untuk ukuran perangkat lunak bersama dengan aturan penghitungan telah diadaptasi beberapa kali, dataset yang digunakan albrecht dengan hasil FP 73.33

Seleksi fitur, baru-baru ini digunakan di bidang rekayasa perangkat lunak untuk meningkatkan akurasi, model biaya perangkat lunak. Ide di balik memilih subset paling informatif fitur yang tersedia berasal dari hipotesis bahwa mengurangi dimensi dataset secara signifikan akan mengurangi kompleksitas dan waktu yang diperlukan untuk mencapai perkiraan menggunakan teknik pemodelan tertentu.

3 METODE YANG DIUSULKAN

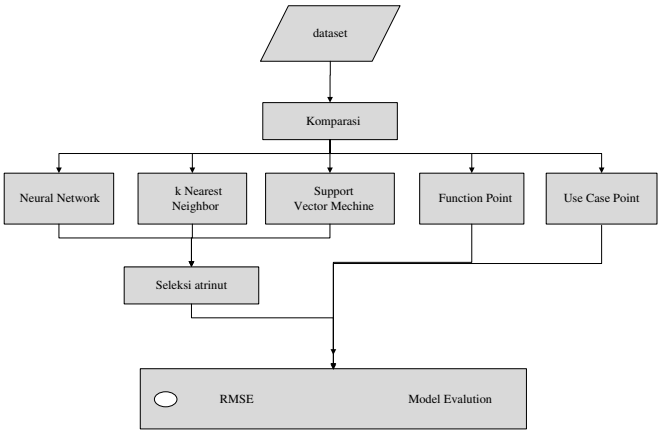
Peneliti mengusulkan untuk mengkomparasi metode machine learning (kNN, SVM, NN) dan non machine learning (FP dan UCP) dan menambahkan algoritma seleksi atribut (Forward Selection dan Backward Elimination). Gambar 1 menunjukan komparasi algoritma klasifikasi dan seleksi atribut yang diusulkan

Model yang diusulkan dalam penelitian ini mulai dari pengolahan dataset hingga menghasilkan model dan selanjutnya model akan diuji dengan menggunakan dataset testing. Dataset akan dilakukan pengujian dengan menggunakan algoritma berbeda secara bergantian. Hasil dari pengujian ini akan dilakukan komparasi algoritma mana yang

Copyright © 2015 IlmuKomputer.Com

paling akurat. Langkah-langkah tersebut akan dilakukan sebagai berikut:

1 Dataset untuk komparasi dan Model yang terbentuk akan langsung diuji dengan dataset testing yang terbentuk, dan nilai akurasi model akan dirata-ratakan.
2 Membandingkan hasil performa antara algoritma *support vector machine, k- nearest neighbor, neural network, function point* dan *use case point.* Hasil RMSE metode machine learning dan metode non-machine learning ini akan dibandingkan. Nilai RMSE yang paling rendah merupakan metode paling akurat.
3 Dilanjutkan dengan menambahkan seleksi atribut pada metode yang paling akurat.



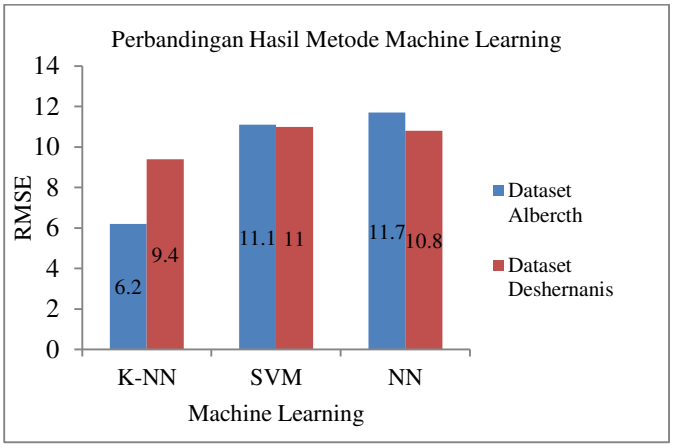
Gambar 1. Komparasi metode machine learning dan non machine learning serta seleksi atribut.

4 HASIL EKSPERIMEN

Penelitian yang dilakukan menggunakan komputer dengan sistem operasi Microsoft Windows 7 Professional 64-bit. Aplikasi yang digunakan adalah RapidMiner 5.2. Data penelitian ini menggunakan dataset Albercth dan dataset Desherhanis.

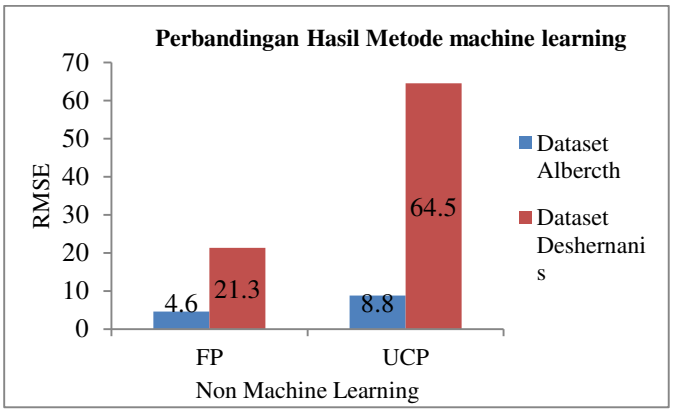
Pada penelitian pertama yang menggunakan metode machine learning kNN, NN dan SVM, dapat dilihat hasil komparasi RMSE pada Gambar 2. Dari Gambar 2 dapat kita ketahui bahwa K-NN mempunyai nilai RSME yang paling baik, dengan nilai RMSE 6.2 untuk dataset Albercth dan 9.4 untuk dataset Desherhanis. NN menghasilkan RMSE 11.7 poin untuk dataset Albercth dan 10.8 untuk dataset Desherhanis . SVM menghasilkan RMSE 11.1 untuk dataset Albercth dan 11 untuk dataset Desherhanis. Dari hasil tersebut, maka dapat kita simpulkan kNN merupakan metode machine learning yang baik. Gambar 2 merupakan grafik komparasi RMSE metode machine learning.

Dari penelitian dari Wen (Wen, Li, Lin, Hu, & Huang, 2012a), Nassif (Nassif, Capretz, & Ho, 2011), dan Kocaguneli (Kocaguneli & Menzies, 2013), juga melakukan penelitian dihasilkan hal yang berbeda hal ini dikerenakan beberapa faktor yang yang mempengaruhi dari hasil tersebut salah satu peneliti Wen (Wen et al., 2012), juga mendapatkan hasil untuk nilai estimasi usaha perangkat lunak untuk metode machine learning adalah kNN.



Gambar 2 Grafik Komparasi RMSE Metode Machine Learning

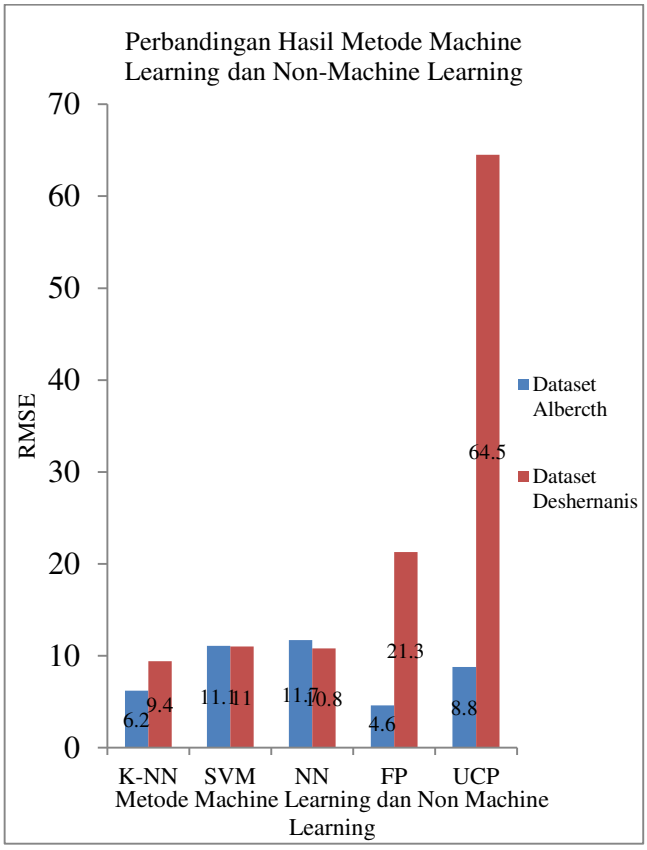
Pada penelitian kedua yang menggunakan metode non-machine learning FP dan UCP, dapat dilihat hasil komparasi RMSE pada Gambar 3. Dari Gmabar 3 dapat ketahui bahwa FP mempunyai nilai RSME yang paling baik, dengan nilai RMSE 4.6 poin untuk dataset Albercth dan 21.3 poin untuk dataset Desherhanis. UCP menghasilkan RMSE 8.8 poin untuk dataset Albercth dan 64.3 poin untuk dataset Desherhanis. Dari hasil tersebut, maka dapat kita simpulkan FP merupakan metode non-machine learning yang baik. Gambar 3 merupakan grafik komparasi RMSE metode non- machine learning.



Gambar 3. Grafik Komparasi RMSE Metode Non-Machine Learning

Penelititan ketiga adalah komparasi hasil metode machine learning dan non-machine learning terbaik. Pada penelitian pertama, dikomparasikan tiga metode machine learning yaitu kNN, NN dan SVM. Didapatkan hasil terbaik adalah kNN. Pada penelitian kedua, dikomparasikan dua metode non machine learning yaitu FP dan UCP. Didapatkan hasil terbaik adalah kNN. Maka pada penelitian ketiga ini dikomparasikan antara kNN dan FP dan menghasilkan kNN sebagai metode terbaik untuk estimasi perangkat lunak. Gambar 4 merupakan hasil komparasi metode machine learning dan non machine learning terbaik untuk estimasi perangkat lunak

Copyright © 2015 IlmuKomputer.Com



Gambar 4. Perbandingan Hasil Metode Machine Learning dan Non-Machine Learning

Penelitian keempat dapat dilihat pada Tabel 1 merupakan tabel pengujian seleksi atribut terbaik, dapat dianalisa untuk seleksi atribut terbaik untuk setiap algoritma klasifikasi. Untuk kNN, lebih baik menggunakan *feature* backward elimination dan forward selection.

Metode seleksi atribut backward elimination dan forward selection merupakan alternative untuk mengurangi kemungkinan adanya multikolinearitas dalam model yang dihasilkan. Prosedur ini tidak selalu mengarahkan ke model yang terbaik, mengingat kita hanya mempertimbangkan sebuah subset kecil dari semua model-model yang mungkin. Sehingga resiko melewatkan atau kehilangan model terbaik akan bertambah seiring dengan penambahan jumlah variabel bebas.

Tabel 1. Pengujian Seleksi Aribut Untuk kNN

Seleksi Atribut Untuk K-NN		
	Backward Elimination	Forward Selection
Dataset Albercth	7.5	2.5
Dataset Deshernanis	7.5	6.4

Berdasarkan metode seleksi atribut dapat dibedakan menjadi dua tipe, yaitu *filter* dan *wrapper* [17]. Contoh dari tipe *filter* adalah *information gain* (IG), *chi-square*, dan *log likelihood ratio*. Contoh dari tipe *wrapper* adalah *forward selection* dan *backward elimination* [18]. Hasil akurasi dari tipe *wrapper* lebih tinggi daripada tipe *filter*, tetapi hasil ini tercapai dengan tingkat kompleksitas yang besar. Tabel 3 dan hasil penelitian dapat disimpulkan bahwa seleksi atribut FS mendapatkan hasil yang terbaik. Dengan menggunakan kNN untuk dua dataset baik dataset Albercth dan dataset Deshernanis, algoritma seleksi atribut FS mendapatkan hasil yang paling baik untuk digunakan pada estimasi usaha perangkat lunak.

5 KESIMPULAN

SVM didapatkan kNN dengan hasil terbaik dengan nilai RMSE yang paling baik, dengan nilai RMSE 6.2 untuk dataset Albercth dan 9.4 untuk dataset Desherhanis. Hasil dari metode non ML antara FP dan UCP didapatkan FP mempunyai nilai RSME yang paling baik, dengan nilai RMSE 4.6 untuk dataset Albercth dan 21.3 untuk dataset Desherhanis. Dapat disimpulkan hasil dari komparasi antara kNN dan FP dan menghasilkan kNN sebagai metode terbaik untuk estimasi perangkat lunak.

Seleksi atribut FS mendapatkan hasil yang terbaik. Dengan menggunakan kNN untuk dua dataset baik dataset Albercth dan dataset Deshernanis, algoritma Seleksi Atribut FS mendapatkan hasil yang paling baik untuk digunakan pada estimasi usaha perangkat lunak.menghasilkan kNN dengan Seleksi Atribut FS sebagai metode terbaik untuk estimasi perangkat lunak.

REFERENSI

Albrecht, A. J., & Gaffney, J. E. (1983). Software Function, Source Lines of Code, and Development Effort Prediction: A Software Science Validation. *IEEE Transactions on Software Engineering*, *SE-9*(6), 639–648. doi:10.1109/TSE.1983.235271

Boehm, B. W., & Papaccio, P. N. (1988). Understanding and controlling software costs. *IEEE Transactions on Software Engineering*. doi:10.1109/32.6191

Choy, S. K., Tang, M. L., & Tong, C. S. (2011). Image segmentation using fuzzy region competition and spatial/frequency information. *IEEE Transactions on Image Processing : A Publication of the IEEE Signal Processing Society*, *20*(6), 1473–84. doi:10.1109/TIP.2010.2095023

Danger, R., Segura-Bedmar, I., Martínez, P., & Rosso, P. (2010). A comparison of machine learning techniques for detection of drug target articles. *Journal of Biomedical Informatics*, *43*(6), 902–13. doi:10.1016/j.jbi.2010.07.010

Dennis, A. (2012). *Systems Analysis and Design: An Applied Approach*. Wiley; 5 edition (January 18, 2012). Retrieved from http://www.philadelphia.edu.jo/it/cs/syllabus/731332.pdf

Dubois, S., Rasovska, I., & De Guio, R. (2009). Towards an automatic extraction of Generalized System of Contradictions out of solutionless Design of Experiments. In *3nd IFIP Working Conference on Computer Aided Innovation (CAI): Growth and Development of CAI*. doi:10.1007/978-3-642-03346-9

El-Sebakhy, E. a. (2011). Functional networks as a novel data mining paradigm in forecasting software development efforts. *Expert Systems with Applications*, *38*(3), 2187–2194. doi:10.1016/j.eswa.2010.08.005

Ian Sommerville. (2011). *Software engineering 9. New York 1992*. Addison-Wesley; 9 edition (March 13, 2010). doi:10.1109/MC.1987.1663532.

Kocaguneli, E., & Menzies, T. (2013). Software effort models should be assessed via leave-one-out validation. *Journal of Systems and Software*, *86*(7), 1879–1890. doi:10.1016/j.jss.2013.02.053

Mehmood, A., S. Palli, A., & Khan, M. N. A. (2014). A Study of Sentiment and Trend Analysis Techniques for Social Media Content. *International Journal of Modern Education and Computer Science*, *6*(December), 47–54. doi:10.5815/ijmecs.2014.12.07

Nassif, A. B., Capretz, L. F., & Hill, R. (1993). A Regression Model with Mamdani Fuzzy Inference System for Early Software Effort Estimation Based on Use Case Diagrams, 615–620.

Nassif, A. B., Capretz, L. F., & Ho, D. (2011). Estimating Software Effort Based on Use Case Point Model Using Sugeno Fuzzy

Copyright © 2015 IlmuKomputer.Com

Inference System. *2011 IEEE 23rd International Conference on Tools with Artificial Intelligence*, 393–398. doi:10.1109/ICTAI.2011.64

Nassif, A. B., Capretz, L. F., & Ho, D. (2012). Estimating Software Effort Using an ANN Model Based on Use Case Points. *2012 11th International Conference on Machine Learning and Applications*, *7*, 42–47. doi:10.1109/ICMLA.2012.138

Nunes, N., Constantine, L., & Kazman, R. (2011a). IUCP: Estimating interactive-software project size with enhanced use-case points. *IEEE Software*, *28*, 64–73. doi:10.1109/MS.2010.111

Nunes, N., Constantine, L., & Kazman, R. (2011b). IUCP: Estimating interactive-software project size with enhanced use-case points. *IEEE Software*. doi:10.1109/MS.2010.111

Oliveira, A. L. I., Braga, P. L., Lima, R. M. F., & Cornélio, M. L. (2010). GA-based method for feature selection and parameters optimization for machine learning regression applied to software effort estimation. *Information and Software Technology*, *52*(11), 1155–1166. doi:10.1016/j.infsof.2010.05.009

Oliveira, A. L. I., Braga, P. L., Lima, R. M. F., & Cornélio, M. L. (2010). GA-based method for feature selection and parameters optimization for machine learning regression applied to software effort estimation. *Information and Software Technology*, *52*(11), 1155–1166. doi:10.1016/j.infsof.2010.05.009

Papatheocharous, E., & Andreou, A. S. (2009). Hybrid Computational Models for Software Cost Prediction : An Approach Using Artificial Neural Networks and Genetic Algorithms, 87–100.

Shepperd, M., & MacDonell, S. (2012, August). Evaluating prediction systems in software project estimation. *Information and Software Technology*. Elsevier B.V. doi:10.1016/j.infsof.2011.12.008

Wang, S., Li, D., Song, X., Wei, Y., & Li, H. (2011). A feature selection method based on improved fisher's discriminant ratio for text sentiment classification. *Expert Systems with Applications*, *38*(7), 8696–8702. doi:10.1016/j.eswa.2011.01.077

Wang, W., & Zhou, Z. H. (2012). Learnability of multi-instance multi-label learning. *Chinese Science Bulletin*, *57*, 2488–2491. doi:10.1007/s11434-012-5133-z

BIOGRAFI PENULIS

Ega Kartika Adhitya. Menyelesaikan pendidikan S1 Perikanan di Universitas Diponegoro, Semarang, S2 Magister Teknik Informatika di Universitas Dian Nuswantoro Semarang. Saat ini menjadi Mahasiswa di Semarang. Minat penelitian saat ini adalah software engineering.

Romi Satria Wahono. Memperoleh gelar B.Eng dan M.Eng pada bidang ilmu komputer di Saitama University Japan, dan Ph.D pada bidang software engineering di Universiti Teknikal Malaysia Melaka. Pengajar dan peneliti di Fakultas Ilmu Komputer, Universitas Dian Nuswantoro. Pendiri dan CEO PT Brainmatics, perusahaan yang bergerak di bidang pengembangan software. Minat penelitian pada bidang software engineering dan machine learning. Profesional member dari asosiai ilmiah ACM, PMI dan IEEE Computer Society.

Hendro Subagyo. Menyelesaikan program S1 (B.Eng) dan S2 (M.Eng) pada jurusan Ilmu Komputer dan Informasi Matematik di The University of Electro-Communications, Tokyo, Jepang pada tahun 1999 dan 2001. Di Indonesia berstatus sebagai peneliti di Pusat Dokumentasi Informasi Ilmiah. Lembaga Ilmu Pengetahuan. Memiliki minat pada sistem operasi, pemrograman dan bahasa pemrograman (khususnya Java dan Real-Time Java) dan komputer aritmatika. Tema penelitian saat ini adalah software engineering dan ubiquitous computing.

Copyright © 2015 IlmuKomputer.Com