

Hasil pada Tabel 7 hingga Tabel 9 telah menunjukkan bahwa di ketiga proyek perangkat lunak yang diuji, pengiriman proyek perangkat lunak terbaik dicapai oleh proyek Spring. Angka-angka pada proyek Apache dan JBoss menunjukkan bahwa kedua proyek ini sebagian besar memiliki pengiriman sprint yang kurang baik. Di 347 dan 352 sprint, masing-masing dari dua proyek ini memiliki lebih dari 200 sprint yang kurang berprestasi di setiap tahap perkembangan. Namun, proyek Spring tampil cukup mengagumkan. Dari 476 sprint dan tiga tahap kemajuan, setidaknya lebih dari 300 sprint berhasil memenuhi atau bahkan melampaui target kerja minimum.

4.2 Analisis Hasil Pengujian

Hasil yang terdapat dalam penelitian ini dapat dikaitkan dengan beberapa alasan. Contohnya pengaruh dimensi data, terutama dimensi pada dataset *issues* mungkin menjadi faktor utama yang mempengaruhi performa *prediction model* yang dibangun menggunakan KNN. Kemudian, untuk hasil Decision Tree, hal ini mungkin terkait dengan ketidakmampuannya untuk menerapkan *regression* serta kompleksitas fitur data yang dimiliki mungkin menghambat performa keseluruhannya di ketiga tahap kemajuan *software project* JBoss dan Spring.

Perlu dicatat bahwa hasil yang diperoleh pada penelitian ini diperoleh melalui pendekatan yang lebih sederhana berdasarkan penelitian sebelumnya [7]. Meskipun dataset yang digunakan memang berasal dari studi [7] yang serupa, ditunjukkan oleh data *software project* yang kami gunakan dalam penelitian ini, kendala yang disebutkan sebelumnya mengenai kurangnya *issue dependencies* berpengaruh pada hasil. Oleh karena itu, metode yang berbeda harus diterapkan untuk membangun model prediksi yang kami sarankan. Dengan demikian, hasil pengujian tidak dapat dibandingkan secara langsung karena perbedaan kondisi yang terjadi antara penelitian ini dengan penelitian sebelumnya.

5. Kesimpulan

Pada penelitian ini, kami mampu mengimplementasikan *prediction model* dalam lingkungan *iterative-based software development*. Model dibuat menggunakan tiga metode berupa *Random Forest Regressor*, *KNN*, dan *Decision Tree Regressor*. Hasil dari pengujian yang dilakukan telah menunjukkan bahwa *Random Forest Regressor* merupakan algoritma dengan performa terbaik. Selain itu dalam pembuatan model, didapatkan performa setiap *sprint* dalam ketiga *software project* yang diuji yaitu Apache, JBoss, dan Spring. Melalui hasil pencapaian setiap *sprint* dari ketiga *software project* tersebut, ditemukan bahwa sebagian besar dari *sprint* dalam proyek Spring mampu memenuhi dan/atau melampaui target minimum. Hal ini menunjukkan bahwa proyek Spring merupakan *software project* dengan tingkat *sprint delivery* terbaik.

Dalam pengujian terdapat beberapa hasil dari *KNN* dan *Decision Tree Regressor* yang mampu melampaui performa *Random Forest Regressor*. Tetapi, performa yang stabil dari *Random Forest Regressor* pada setiap tahapan *sprint* dari semua *software project* yang diuji menunjukkan bahwa algoritma ini lebih baik untuk digunakan pada pembentukan *prediction model*. Oleh sebab itu walaupun tidak meraih hasil optimum, sistem prediksi yang diusulkan memungkinkan untuk diterapkan dalam lingkungan *iterative-based software development*. Sebagai saran ke depannya, penelitian ini bisa dikembangkan lagi untuk meraih hasil optimum. Contohnya dengan menguji performa *prediction model* dari penelitian ini pada data *software project lain*, melakukan optimisasi pada *preprocessing* data dalam penelitian ini, atau mencoba membangun *prediction model* lain menggunakan kombinasi *ensemble learning methods* seperti *Random Forest* dengan *Deep Neural Networks*.

Daftar Pustaka

- [1] L. Gonçalves, "Scrum," *Control. Manag. Rev.*, vol. 62, no. 4, pp. 40–42, May 2018, doi: 10.1007/s12176-018-0020-3.
- [2] F. Hayat, A. U. Rehman, K. S. Arif, K. Wahab, and M. Abbas, "The Influence of Agile Methodology (Scrum) on Software Project Management," *Proc. - 20th IEEE/ACIS Int. Conf. Softw. Eng. Artif. Intell. Netw. Parallel/Distributed Comput. SNPD 2019*, pp. 145–149, Jul. 2019, doi: 10.1109/SNPD.2019.8935813.
- [3] J. Wright, "Scrum: the complete guide to the agile project management framework that helps the software development lean team to efficiently structure and simplify the work & solve problems in half the time." p. 95, 2020.
- [4] M. Hron and N. Obwegeser, "Why and how is Scrum being adapted in practice: A systematic review," *J. Syst. Softw.*, vol. 183, p. 111110, Jan. 2022, doi: 10.1016/J.JSS.2021.111110.
- [5] M. Hron and N. Obwegeser, "Scrum in practice: An overview of Scrum adaptations," *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, vol. 2018-Janua, pp. 5445–5454, 2018, doi: 10.24251/hicss.2018.679.

- [6] M. Marinho, J. Noll, and S. Beecham, "Uncertainty management for global software development teams," *Proc. - 2018 Int. Conf. Qual. Inf. Commun. Technol. QUATIC 2018*, pp. 238–246, Dec. 2018, doi: 10.1109/QUATIC.2018.00042.
- [7] M. Choetkiertikul, H. K. Dam, T. Tran, A. Ghose, and J. Grundy, "Predicting Delivery Capability in Iterative Software Development," *IEEE Trans. Softw. Eng.*, vol. 44, no. 6, pp. 551–573, 2018, doi: 10.1109/TSE.2017.2693989.
- [8] M. Choetkiertikul, H. K. Dam, T. Tran, and A. Ghose, "Predicting delays in software projects using networked classification," *Proc. - 2015 30th IEEE/ACM Int. Conf. Autom. Softw. Eng. ASE 2015*, pp. 353–364, 2016, doi: 10.1109/ASE.2015.55.
- [9] C. Verwijs and D. Russo, "A Theory of Scrum Team Effectiveness." 2021. [Online]. Available: <http://arxiv.org/abs/2105.12439>
- [10] P. Pospieszny, B. Czarnacka-Chrobot, and A. Kobylinski, "An effective approach for software project effort and duration estimation with machine learning algorithms," *J. Syst. Softw.*, vol. 137, pp. 184–196, Mar. 2018, doi: 10.1016/J.JSS.2017.11.066.
- [11] M. Choetkiertikul, H. K. Dam, T. Tran, T. Pham, A. Ghose, and T. Menzies, "A Deep Learning Model for Estimating Story Points," *IEEE Trans. Softw. Eng.*, vol. 45, no. 7, pp. 637–656, 2019, doi: 10.1109/TSE.2018.2792473.
- [12] P. Ardimento and C. Mele, "Using BERT to Predict Bug-Fixing Time," *IEEE Conf. Evol. Adapt. Intell. Syst.*, vol. 2020-May, May 2020, doi: 10.1109/EAIS48028.2020.9122781.
- [13] M. Choetkiertikul, H. K. Dam, T. Tran, and A. Ghose, "Characterization and prediction of issue-related risks in software projects," *IEEE Int. Work. Conf. Min. Softw. Repos.*, vol. 2015-Augus, pp. 280–291, 2015, doi: 10.1109/MSR.2015.33.
- [14] D. Denisko and M. M. Hoffman, "Classification and interaction in random forests," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 115, no. 8, pp. 1690–1692, Feb. 2018, doi: 10.1073/PNAS.1800256115.
- [15] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 8, no. 4, p. e1249, Jul. 2018, doi: 10.1002/WIDM.1249.
- [16] S. Zhang, M. Zong, X. Zhu, D. Cheng, and X. Li, "Learning k for kNN classification," *ACM Trans. Intell. Syst. Technol.*, vol. 8, no. 43, 2017, doi: 10.1145/2990508.
- [17] H. Al-Shehri *et al.*, "Student performance prediction using Support Vector Machine and K-Nearest Neighbor," *Can. Conf. Electr. Comput. Eng.*, Jun. 2017, doi: 10.1109/CCECE.2017.7946847.
- [18] R. Hasan, S. Palaniappan, A. R. A. Raziff, S. Mahmood, and K. U. Sarker, "Student Academic Performance Prediction by using Decision Tree Algorithm," *2018 4th Int. Conf. Comput. Inf. Sci. Revolutionising Digit. Landsc. Sustain. Smart Soc. ICCOINS 2018 - Proc.*, Oct. 2018, doi: 10.1109/ICCOINS.2018.8510600.
- [19] A. K. Hamoud, A. S. Hashim, and W. A. Awadh, "Predicting Student Performance in Higher Education Institutions Using Decision Tree Analysis," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 5, no. 2, p. 26, 2018, doi: 10.9781/ijimai.2018.02.004.