Jurnal Informatika dan Rekayasa Perangkat Lunak ISSN 2656-2855
Vol. 7, No. 2, September 2025, Hal. 244-250 e-ISSN 2685-5518

Perbandingan Apache Airflow dan Apache Spark dalam Proses ETL untuk Memprediksi *DropOut* dan Keberhasilan Akademik Mahasiswa

Triyan Agung Laksono[1*], Widyastuti Andriyani[2]
[1,2]Magister Teknologi Informasi, Universitas Teknologi Digital Indonesia.
[*]e-mail: [1]triyan31@stiesbi.ac.id, [1]widya@utdi.ac.id

Abstrak

Prediksi putus sekolah di pendidikan tinggi menjadi penting karena berdampak pada keberhasilan akademik mahasiswa dan efektivitas institusi pendidikan secara keseluruhan. Penelitian ini bertujuan membangun pipeline ETL otomatisasi menggunakan Apache Airflow dan Apache Spark untuk memproses data akademik dan memprediksi status kelulusan mahasiswa. Dataset yang digunakan terdiri dari 4.424 sample dengan 36 fitur yang mencakup atribut demografi, akademik, dan sosial ekonomi. Data diproses melalui tahapan ekstraksi, transformasi(termasuk normalisasi SMOTE), dengan pemuatan ke dalam model Random Forest. Hasil evaluasi menunjukkan akurasi 62,93% dan nilai ROC-AUC tertinggi 0,81 untuk kelas dropout. Pipeline Airflow unggul dalam efisiensi penjadwalan tugas, sedangkan Spark efektif untuk pemrosesan data berskala besar. Pendekatan ini menunjukkan potensi praktis dalam mendukung sistem peringatan dini untuk pengambilan keputusan kebijakan akademik. Penelitian ini memberikan kontribusi dalam intergasi teknologi big data dan machine learning untuk pengolahan data pendidikan tinggi secara efisien dan terotomatisasi.

Kata kunci: *Apache Airflow, Apache Spark, ETL, prediksi dropout, machine learning.*

Abstrak

Dropout prediction in higher education is important because it impacts the academic success of students and the overall effectiveness of educational institutions. This research aims to build an automated ETL pipeline using Apache Airflow and Apache Spark to process academic data and predict student graduation status. The dataset used consists of 4,424 samples with 36 features covering demographic, academic, and socio-economic attributes. The data is processed through the stages of extraction, transformation (including SMOTE normalization), with loading into the Random Forest model. The evaluation results showed an accuracy of 62.93% and the highest ROC-AUC value of 0.81 for the dropout class. The Airflow pipeline excels in task scheduling efficiency, while Spark is effective for large-scale data processing. This approach shows practical potential in supporting early warning systems for academic policy decision-making. This research contributes to the intergation of big data and machine learning technologies for efficient and automated higher education data processing.

Keywords: *Apache Airflow, Apache Spark, ETL, dropout prediction, machine learning*

PENDAHULUAN

Dalam era saat ini, banyak universitas yang mengandalkan data untuk membuat keputusan strategis dan operasional. Data akademik, yang meliputi informasi pendaftaran mahasiswa, nilai, dan aktivitas pendidikan mahasiswa, terus bertambah seiring berjalannya waktu, sehingga membutuhkan sistem manajemen data yang cepat, efisien, dan terukur (Giovanelli *et al.*, 2022; Pogiatzis and Samakovitis, 2020). Proses Extract, Transform, Load (ETL) penting dalam mengelola data berskala besar, terutama di lingkungan big-data yang kompleks. Namun, sistem ETL, terutama ETL tradisional, sering kali tidak dapat menangani data berskala besar dan berbagai jenis data dengan efisiensi yang tinggi (Gueddoudj and Chikh, 2023; Mhon and Kham, 2020). Sistem ETL konvensional menghadapi kendala yang signifikan dalam hal skalabilitas dan waktu pemrosesan saat menangani data berskala besar (Lee and Park, 2021). Pendekatan berbasis *big data* telah diimplementasikan di berbagai bidang bisnis dan akademis untuk mengatasi kendala skalabilitas, seperti prediksi churn di perusahaan (Zdravevski *et al.*, 2020).

Dengan perkembangan teknologi yang terus berkembang pesat saat ini, tantangan tersebut dapat diatasi dengan teknologi modern seperti Apache Spark dan Apache Airflow. Apache Spark dan Apache Airflow telah banyak digunakan dalam membangun *pipeline* data yang efisien dan otomati(Gueddoudj and Chikh, 2023; Mitchell *et al.*, 2019). Apache Spark sendiri telah menawarkan pemrosesan data paralel yang sangat cepat dengan dukungan eksekusi

terdistribusi, sedangkan Apache Airflow menyediakan orkestrasi *pipeline* yang memungkinkan eksekusi tugas yang terjadwal dan terstruktur serta mendukung manajemen ketergantungan tugas yang fleksibel(Mohit Nara *et al.*, 2023; Singh *et al.*, 2021; Stan *et al.*, 2019). Penggunaan virtualisasi adaptif dalam *pipeline* ETL berbasis *cloud* juga telah terbukti meningkatkan fleksibilitas dan efisiensi dalam pemrosesan data secara real-time(Abdelhamid *et al.*, 2023). Penelitian sebelumnya telah menunjukkan bahwa penggunaan airflow Apache dalam *pipeline* ETL dapat meningkatkan fleksibilitas dalam penjadwalan dan pengelolaan ketergantungan tugas(Mitchell *et al.*, 2019). Selain itu, otomatisasi *pipeline* berbasis ML dapat mempercepat penerapan dan meningkatkan efisiensi pengelolaan alur kerja data berskala besar(Ramanan *et al.*, 2020). Penelitian lain telah membuktikan bahwa Apache Spark mampu memberikan performa yang lebih baik dibandingkan sistem pemrosesan data lainnya di lingkungan *big data*(Gueddoudj and Chikh, 2023; Gulino *et al.*, 2020; Singh *et al.*, 2021).

Studi ini bertujuan untuk mengembangkan *pipeline* ETL yang terukur dan efisien menggunakan Apache Spark dan Apache Airflow untuk data akademik dari universitas. *Pipeline* yang dirancang akan mampu menangani data yang heterogen, seperti data demografi mahasiswa, informasi keuangan, dan data akademik, yang telah diproses secara otomatis dan terjadwal. *Dataset* yang digunakan dalam penelitian ini adalah *dataset* yang memiliki 4424 *instance* dan 36 fitur, yang dapat mencakup informasi pendaftaran mahasiswa, latar belakang sosial ekonomi, dan hasil akademik mahasiswa selama dua semester pertama. *Dataset* yang digunakan diambil dari UCI *Machine learning* Repository yang telah dirancang untuk memprediksi tiga kategori utama: putus sekolah, terdaftar, dan lulus di akhir masa studi(Realinho Valentim and Baptista, 2021). Dengan mengadopsi pendekatan berdasarkan kontainerisasi dan orkestrasi otomatis, *pipeline* ini diharapkan dapat mengatasi kendala sistem ETL tradisional dan meningkatkan efisiensi pemrosesan data di lingkungan pendidikan tinggi.

Kontribusi utama dari penelitian ini adalah pengembangan *pipeline* ETL berbasis Apache Spark dan Apache Airflow yang didesain untuk menangani *big data* dan memberikan performa yang tinggi serta efisiensi yang lebih baik dibandingkan dengan pendekatan tradisional. Kemudian penelitian ini mengevaluasi kinerja *pipeline* terhadap waktu pemrosesan dan skalabilitas dan *pipeline* yang dikembangkan diaplikasikan pada data akademik dengan menggunakan studi kasus di perguruan tinggi, untuk menunjukkan keunggulannya dalam mengelola data akademik yang kompleks dan heterogen. Penelitian serupa mengenai prediksi *dropout* mahasiswa menggunakan data mining dan pemodelan berbasis sistem rekomendasi *machine learning* (ML) telah dilakukan oleh beberapa peneliti sebelumnya (Ardchir *et al.*, 2020; Del Bonifro *et al.*, 2020; Oliveira *et al.*, 2019). Namun, penelitian ini menawarkan pendekatan baru untuk membangun *pipeline* ETL end-to-end dengan mengintegrasikan teknologi modern dan mengutamakan efisiensi dan skalabilitas.

TINJAUAN PUSTAKA

ETL (Extract, Transform, Load) merupakan proses penting dalam pengolahan data yang digunakan untuk mengumpulkan data dari berbagi sumber, mengubahnya menjadi format yang seusai, dan memuatnya ke dalam sistem penyimpanan untuk analisis lebih lanjut. Dalam konteks pendidikan yang di mana data akademik yang beragam perlu diintegrasikan dan dianalisis, sehingga proses ETL menjadi sangat krusial untuk mendukung pengambilan keputusan yang tepat(Gil *et al.*, 2021; Mhon and Kham, 2020).

Ketika berbicara tentang teknologi ETL, ada dua platform populer yang sering digunakan yaitu Apache Airflow dan Apache Spark. Apache Airflow merupakan platform orkestrasi alur kerja yang memungkinkan penggunaan untuk menjadwalkan dan mengelola tugas-tugas ETL secara efisien(Giovanelli et al., 2022). Kelebihan dari Airflow terletak pada kemampuannya untuk mengelola ketergantungan tugas dan memberikan visibilitas yang jelas terhadap alur kerja yang sedang berjalan. Sedangkan Apache Spark merupakan *framework* pemrosesan data yang dirancang untuk menangani data besar dengan kecepatan tinggi(Ahmed *et al.*, 2020). Spark sendiri menawarkan kemampuan pemrosesan pararel dan mendukung berbagi bahasa

pemrograman, sehingga menjadi pilian yang populer untuk analisis data.

Beberapa penelitian telah membahas penggunaan ETL dalam konteks pendidikan. Sebagai contoh penelitian (Gil *et al.*, 2021) yang mengembangkan model berbasis data untuk memprediksi kesuksesan akademik mahasiswa, sedangkan (Zdravevski *et al.*, 2020) menerapkan kerangka kerja ETL berbasis *cloud* untuk digunakan sebagai analisis bisnis dengan fokus pada prediksi churn. Akan tetapi, perbandingan langsung antara Apache Airflow dan Apache Spark dalam konteks pendidikan masih sedikit. Meskipun demikian banyak penelitian tentang ETL dan analisis data, masih kurangnya penelitian yang secara langsung membandingkan kinerja kedua teknologi ini dalam konteks memprediksi mahasiswa yang putus sekolah dan keberhasilan akademik. Penelitian ini bertujuan untuk mengisi kesenjangan tersebut dengan melakukan analisis mendalam terhadap kedua teknologi tersebut dalam konteks yang relevan.

Tidak seperti penelitian sebelumnya yang berfokus pada prediksi dropout yang menggunakan model pembelajaran mesin (Asha *et al.*, 2020; Del Bonifro *et al.*, 2020) atau pada optimalisasi ETL tanpa latar belakang pendidikan (Gueddoudj and Chikh, 2023; Pogiatzis and Samakovitis, 2020), penelitian ini secara khusus mengintegrasikan Apache Airflow dan Apache Spark untuk membangun pipeline ETL otomatis dalam konteks pendidikan tinggi. Dengan demikian, penelitian ini mengisi celah dengan menyajikan analisis perbandingan dua teknologi big data dalam memprediksi risiko putus sekolah mahasiswa, yang belum di teliti secara mendalam sebelumnya.

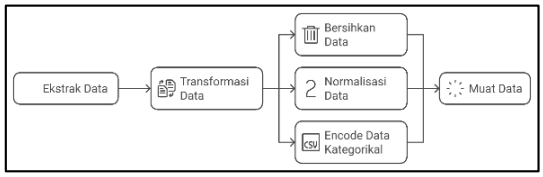
METODE

Penelitian ini menggunakan pendekatan berbasis ML yang di integrasikan dengan *pipeline* otomatis menggunakan Apache Airflow dan Apache Spark untuk memprediksi *dropout* dan keberhasilan mahasiswa. *Dataset* yang digunakan adalah "Predict Students' *Dropout* and Academic Success," yang diambil dari UCI *Machine learning* Repository (Realinho Valentim and Baptista, 2021). *Dataset* berisi data sample 4.424 dengan fitur 36, termasuk atribut demografi, akademik, dan sosial ekonomi mahasiswa.

Dataset yang digunakan dalam penelitian ini mencakup beberapa atribut demografi, akademik, dan sosial ekonomi mahasiswa dari berbagi universitas. *Dataset* dibagi menjadi dua bagian: 80% untuk data pelatihan dan 20% untuk data pengujian(Asha et al., 2020; Krüger et al., 2023; Palacios et al., 2021).

3.1 Proses ETL

Proses ETL bertujuan untuk mempersiapkan data agar dapat digunakan dalam penelitian model ML. Langkah-langkah proses ETL disajikan dalam bentuk pada gambar 1.



Gambar 1. Ilustrasi umum pemrosesan data dari ekstraksi hingga pemuatan data setelah prapemrosesan

Seperti yang telah di sajikan dalam gambar 1 proses ETL terdiri dari beberapa langkah:

a. Ekstraksi data: Mengambil data dari sumber (*dataset*) yang tersedia, *dataset* dari UCI *Machine learning* Repository (Realinho Valentim and Baptista, 2021).
b. Transformasi data dimulai dengan pembersihan nilai hilang, normalisasi numerik menggunakan standar scaler, dan pengkodena fitur kategorikal dengan *one-hot encoding*(Giovanelli et al., 2022).
c. Penyeimbangan kelas menggunakan metode *SMOTE* (Synthetic Minority *Oversampling* Technique).
d. Pemuatan data kedalam *pipeline* ML. Proses ETL diotomatisisasi menggunakan Apache Airflow dengan DAG (Directed Acyclic Graph), sementara transformasi data berskala besar dijalankan menggunakan Apache Spark untuk efisiensi pemerosesan (Ahmed et al., 2020; Mohit Nara et al., 2023).

Selain itu, untuk menangani ketidakseimbangan kelas pada *dataset* yang digunakan, dapat menggunakan teknik *SMOTE* bertujuan untuk meningkatkan jumlah sample pada kelas minoritas, yaitu

dropout dan enrolled. Teknik *SMOTE* ini sudah terbukti sangat efektif untuk meningkatkan performa model pada *dataset* yang tidak seimbang(Ardchir et al., 2020; Gulino et al., 2020).

3.2 Pengembangan dan Pelatihan Model

Model yang digunakan dalam penelitian ini adalah *Random Forest* Classifier dari pustaka sklearn. Algoritma ini dipilih karena dapat menangani *dataset* dengan jumlah fitur yang besar dan dapat menangani overfitting dengan menggunakan metode ensembel (Alyahyan and Düştegör, 2020; Jacob and Henriques, 2023). Model dilatih menggunakan parameter class_weight = 'balanced' untuk mengatasi ketidakseimbangan kelas yang tersisa setelah proses *oversampling* dengan *SMOTE*. Penyesuaian hyperparameter dilakukan menggunakan *GridSearchCV* dengan validasi silang 5 kali lipat. Parameter yang diatur adalah n_estimator, max_depth, dan min_samples_split. Kombinasi terbaik yang didapatkan adalah n_estimator = 200, max_depth = 20, dan min_samples_split = 5, dengan metrik evaluasi utama adalah F1-score berbobot.

3.3 Evaluasi Model

Evaluasi dilakukan dengan menggunakan matriks klasifikasi yaitu akurasi, presisi, recall, F1-soce, dan *ROC-AUC*(Cawi et al., 2019). Evaluasi dilakukan terhadap tiga kelas target untuk menilai ketepatan model secara menyeluruh. Hasil evaluasi ditampilkan dalam bentuk tabel dan grafik ROC untuk masing-masing kelas.

3.4 Implementasi *Pipeline* Otomatis

Pipeline otomatis dirancang menggunakan Apache Airflow dan dijalankan secara terjadwal dengan interval harian(@dialy). *Pipeline* mencakup tugas: Pemuatan model terlatih (final_model.pkl), pembacaan data uji baru (tes_data.csv), prediksi dan pencatatan penggunaan sumber daya (CPU, memori, waktu ekseskusi)(Abdelhamid et al., 2023). Hasil eksekusi menunjukkan *pipeline* berjalan efisien dengan waktu rata-rata 0.69 detik dan penggunaan memori 234.88 MB.

3.5 *Pipeline* Prediksi Menggunakan Apache Spark

Apache Spark digunakan untuk membangun *pipeline* prediksi yang mampu menangani data berskala besar(Aziz *et al.*, 2019; Gulino *et al.*, 2020) Proses meliputi inisialisasi sesi Spark, pemuatan data CSV, konversi DataFrame, pemanggilan model terlatih, prediksi, dan penyimpanan hasil. *Pipeline* menunjukkan waktu eksekusi sekitar 9.57 detik dan penggunaan memori sebesar 420 MB. Pendekatan ini meningkatkan efisiensi dalam skenario *big data* dibandingkan pendekatan konvensional.

Seluruh proses metodologi ini membentuk fondasi *pipeline* prediksi *dropout* mahasiswa yang fleksibel, otomatis, dan efisien, serta dapat direplika oleh institusi pendidikan tinggi lainnya.

HASIL DAN PEMBAHASAN

4.1. hasil Evaluasi Model

Model *Random Forest* yang dikembangkan dievaluasi menggunakan lima metrik utama: akurasi, presisi, recall, F1-score, dan *ROC-AUC*. Hasil pengujian menunjukkan nilai akurasi sebesar 62,93%, presisi sebesar 58,25%, recall sebesar 62,93%, dan F1-score 55,91%.

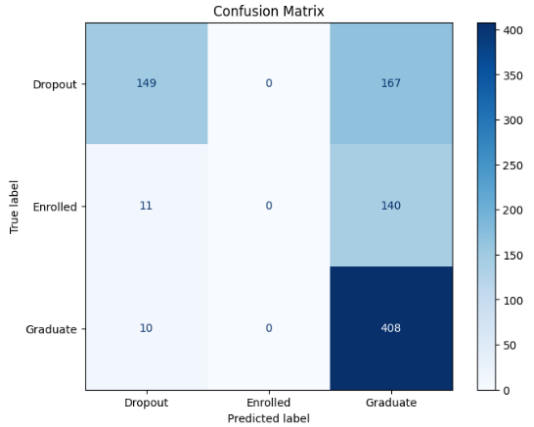
Tabel 1. Hasil Evaluasi Model *Random Forest*

Metrics	Value
Akurasi (Accuracy)	62.93%
Presisi (Precision)	58.25% (weighted)
Recall	62.93% (weighted)
F1-score	55.91% (weighted)

Pada tabel 1, menunjukkan bahwa model memiliki kinerja yang cukup baik dalam memprediksi mahasiswa putus sekolah dan keberhasilan akademik, meskipun masih terdapat beberapa kelemahan pada kelas Enrolled, yang tidak dapat diprediksi dengan baik. Hasil ini menunjukkan performa model yang kompetitif, melebihi studi oleh (Asha *et al.*, 2020) yang melaporkan akurasi 59,2% dalam prediksi *dropout* mahasiswa. Selain itu, *ROC-AUC* 0,81 pada kelas *dropout* lebih tinggi dibandingkan nilai 0,74 pada studi (Del Bonifro et al., 2020).

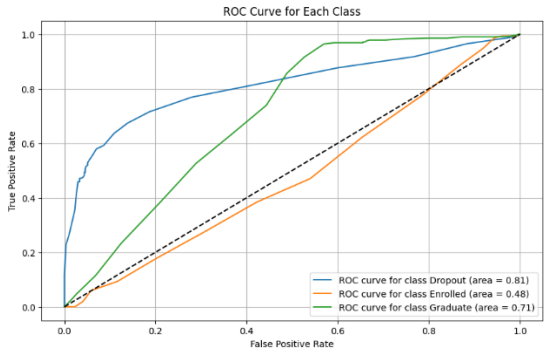
4.2. Analisi *Confusion matrix* Dan *ROC-AUC*

Hasil *confusion matrix* pada gambar 2, menunjukkan bahwa model mampu membedakan kelas "*dropout*" secara relatif lebik baik dibandingkan dengan kelas lain.



Gambar 2. *Confusion matrix*

Analisis *ROC-AUC* untuk masing-masing kelas pada gambar 3, memberikan nilai sebagai berikut: *dropout* = 0.81, graduate = 0.65, dan enrolled = 0.48. Nilai *ROC-AUC* yang rendah pada kelas "enrolled" menunjukkan tangtangan dalam membedakan kategori ini, kemungkinan karena fitur yang tumpang tindih atau distribusi daya yang tidak seimbang.



Gambar 3. Kurva ROC untuk Setiap Kelas

Hal ini sejalan dengan temuan (Krüger et al., 2023), bahwa model prediksi *dropout* memerlukan pendekatan explainable dan sensitif terhadap dinamika kelas minoritas. Kurva ROC yang divisualisasikan pada gambar 3, memperkuat temuan ini dengan memperlihatkan ketidakseimbangan performa antar kelas. Hasil ini menunjukkan perlunya pemilihan fitur tambahan atau teknik balancing yang lebih kuat di masa mendatang.

4.3. Hasil Eksekusi Pipline

Implementasi *pipeline* menggunakan Apache Airflow dan Apache Spark diuji dalam aspek efisiensi dan sumber data yang digunakan. Hasil menunjukkan bahwa *pipeline* Airflow memiliki waktu eksekusi rata-rata 0.69 detik dengan penggunaan memori 234.88 MB, sedangkan Spark membutuhkan waktu 9.57 detik dengan penggunaan memori 420 MB.

Tabel 2. Perbandingan Eksekusi *Pipeline*

Pipeline	**CPU Usage**	**Memory Usage**	**Elapsed Time**
Apache Airflow	263.7%	234.88 MB	0.69 detik
Apache Spark	218%	420.52 MB	9.57 detik

Hasil ini konsisten dengan studi oleh (Ahmed *et al.*, 2020; Aziz *et al.*, 2019) yang menyatakan bahwa Apache Spark unggul dalam pemerosesan pararel namun lebih boros sumber daya dibandingkan pendekatan berbasis orkestrasi seperti Apache Airflow.

4.4. Implikasi Hasil

Hasil ini menunjukkan bahwa model dan *pipeline* yang dikembangkan cukup andal untuk digunakan dalam sistem prediksi droupout mahasiswa. Model memiliki performa klasifikasi yang kompetitif dibandingkan studi sebelumnya, dan *pipeline* mendukung eksekusi otomatis serta skalabel. Temuan ini memiliki implikasi praktis bagi institusi pendidikan dalam membangun sistem peringatan dini berbasis data dan pengambilan keputusan akademik yang lebih tepat sasaran.

SIMPULAN

Penelitian ini berhasil mengembangkan *pipeline* ETL berbasis Apache Airflow dan Apache Spark untuk memprediksi risiko putusan sekolah dan keberhasilan akademik mahasiswa. Model *Random Forest* yang digunakan menunjukkan kinerja klasifikasi yang kompetitif, dengan akurasi 61,93% dan nilai *ROC-AUC* tertinggi 0,81 pada kelas *dropout*. *Pipeline* Airflow menunjukkan efisiensi dalam orkestrasi terjadwal, sementara Spark memberikan keunggulan dalam pemerosesan pararel untuk data bersekala besar.

Kontribusi utama dari penelitian ini adalah implementasi *pipeline* prediktif yang fleksibel dan dapat direplikasi di lingkungan pendidikan tinggi lainnya. Implikasi praktisnya adalah mendukung institusi dalam mengidentifikasi mahasiswa berisiko putus sekolah secara dini, sehingga dapat ditindaklanjuti melalui kebijakan ini dengan integrasi data real-time, pengujian algoritma lain, dan penggunaan data dari berbagi

institusi untuk meningkatkan generalisasi model.

Untuk penelitian selanjutnya, pengembangan dapat diarahkan pada integrasi data real-time dari platform e-learning atau Learning Management System (LMS), penerapan algoritma machine learning yang lebih kompleks seperti XGBoost atau Deep Neural Networks, eksplorasi teknik explainable AI untuk memberikan interpretasi yang lebih transparan terhadap prediksi. Selian itu, perluasan studi dengan melibatkan data dari berbagai institusi pendidikan akan memperkuat generalisasi model dan memungkinkan pengembangan sistem peringatan dini lebih adaptif dan inklusif.

DAFTAR PUSTAKA

Abdelhamid, E., Tsikoudis, N., Duller, M., Sugiyama, M., Marino, N. and Waas, F. (2023), *Adaptive Real-Time Virtualization of Legacy ETL Pipelines in Cloud Data Warehouses*, doi: 10.48786/edbt.2023.64.

Ahmed, N., Barczak, A.L.C., Susnjak, T. and Rashid, M.A. (2020), "A comprehensive performance analysis of Apache Hadoop and Apache Spark for large scale data sets using HiBench", *Journal of Big Data*, Springer International Publishing, Vol. 7 No. 1, doi: 10.1186/s40537-020-00388-5.

Alyahyan, E. and Düştegör, D. (2020), "Predicting academic success in higher education: literature review and best practices", *International Journal of Educational Technology in Higher Education*, Vol. 17 No. 1, p. 3, doi: 10.1186/s41239-020-0177-7.

Ardchir, S., Ouassit, Y., Ounacer, S., Jihal, H., EL Goumari, M.Y. and Azouazi, M. (2020), "Improving Prediction of MOOCs Student Dropout Using a Feature Engineering Approach", pp. 146–156, doi: 10.1007/978-3-030-36653-7_15.

Asha, P., Vandana, E., Bhavana, E. and Shankar, K.R. (2020), "Predicting University Dropout through Data Analysis", *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, IEEE, pp. 852–856, doi: 10.1109/ICOEI48184.2020.9142882.

Aziz, K., Zaidouni, D. and Bellafkih, M. (2019), "Leveraging resource management for efficient performance of Apache Spark", *Journal of Big Data*, Vol. 6 No. 1, p. 78, doi: 10.1186/s40537-019-0240-1.

Del Bonifro, F., Gabbrielli, M., Lisanti, G. and Zingaro, S.P. (2020), "Student Dropout Prediction", pp. 129–140, doi: 10.1007/978-3-030-52237-7_11.

Cawi, E., La Rosa, P.S. and Nehorai, A. (2019), "Designing machine learning workflows with an application to topological data analysis", *PLOS ONE*, Vol. 14 No. 12, p. e0225577, doi: 10.1371/journal.pone.0225577.

Gil, P.D., da Cruz Martins, S., Moro, S. and Costa, J.M. (2021), "A data-driven approach to predict first-year students' academic success in higher education institutions", *Education and Information Technologies*, Vol. 26 No. 2, pp. 2165–2190, doi: 10.1007/s10639-020-10346-6.

Giovanelli, J., Bilalli, B. and Abelló, A. (2022), "Data pre-processing pipeline generation for AutoETL", *Information Systems*, Vol. 108, p. 101957, doi: 10.1016/j.is.2021.101957.

Gueddoudj, E.Y. and Chikh, A. (2023), "Towards a Scalable and Efficient ETL", *International Journal of Computing and Digital Systems*, Vol. 14 No. 1, pp. 10223–10231, doi: 10.12785/ijcds/140195.

Gulino, A., Canakoglu, A., Ceri, S. and Ardagna, D. (2020), "Performance Prediction for Data-driven Workflows on Apache Spark", *2020 28th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, IEEE, pp. 1–8, doi: 10.1109/MASCOTS50786.2020.9285944.

Jacob, D. and Henriques, R. (2023), "Educational Data Mining to Predict Bachelors Students' Success", *Emerging Science Journal*, Vol. 7, pp. 159–171, doi: 10.28991/ESJ-2023-SIED2-013.

Krüger, J.G.C., Britto, A. de S. and Barddal, J.P. (2023), "An explainable machine learning approach for student dropout prediction", *Expert Systems with*

Applications, Vol. 233, p. 120933, doi: 10.1016/j.eswa.2023.120933.

Lee, S. and Park, S. (2021), "Performance Analysis of Big Data ETL Process over CPU-GPU Heterogeneous Architectures", *2021 IEEE 37th International Conference on Data Engineering Workshops (ICDEW)*, IEEE, pp. 42–47, doi: 10.1109/ICDEW53142.2021.00015.

Mhon, G.G.W. and Kham, N.S.M. (2020), "ETL Preprocessing with Multiple Data Sources for Academic Data Analysis", *2020 IEEE Conference on Computer Applications(ICCA)*, IEEE, pp. 1–5, doi: 10.1109/ICCA49400.2020.9022824.

Mitchell, R., Pottier, L., Jacobs, S., Silva, R.F. da, Rynge, M., Vahi, K. and Deelman, E. (2019), "Exploration of Workflow Management Systems Emerging Features from Users Perspectives", *2019 IEEE International Conference on Big Data (Big Data)*, IEEE, pp. 4537–4544, doi: 10.1109/BigData47090.2019.9005494.

Mohit Nara, Aquila Shaikh and Rashmita Pradhan. (2023), "Managing Data Pipeline with Apache Airflow", *International Journal of Advanced Research in Science, Communication and Technology*, pp. 244–250, doi: 10.48175/IJARSCT-12134.

Oliveira, M.M. de, Barwaldt, R., Pias, M.R. and Espindola, D.B. (2019), "Understanding the Student Dropout in Distance Learning", *2019 IEEE Frontiers in Education Conference (FIE)*, IEEE, pp. 1–7, doi: 10.1109/FIE43999.2019.9028433.

Palacios, C.A., Reyes-Suárez, J.A., Bearzotti, L.A., Leiva, V. and Marchant, C. (2021), "Knowledge Discovery for Higher Education Student Retention Based on Data Mining: Machine Learning Algorithms and Case Study in Chile", *Entropy*, Vol. 23 No. 4, p. 485, doi: 10.3390/e23040485.

Pogiatzis, A. and Samakovitis, G. (2020), "An Event-Driven Serverless ETL Pipeline on AWS", *Applied Sciences*, Vol. 11 No. 1, p. 191, doi: 10.3390/app11010191.

Ramanan, B., Drabeck, L., Woo, T., Cauble, T. and Rana, A. (2020), "~PB&J~ - Easy Automation of Data Science/Machine Learning Workflows", *2020 IEEE International Conference on Big Data (Big Data)*, IEEE, pp. 361–371, doi: 10.1109/BigData50022.2020.9378128.

Realinho Valentim, V.M.M.M.J. and Baptista, L. (2021), "Predict Students' Dropout and Academic Success", doi: https://doi.org/10.24432/C5MC89.

Singh, V.K., Karnam, S.E. and Hanji, B.R. (2021), "Orchestration of ML-Based Recommendation Systems", *Journal of University of Shanghai for Science and Technology*, Vol. 23 No. 08, pp. 173–180, doi: 10.51201/JUSST/21/08340.

Stan, C.-S., Pandelica, A.-E., Zamfir, V.-A., Stan, R.-G. and Negru, C. (2019), "Apache Spark and Apache Ignite Performance Analysis", *2019 22nd International Conference on Control Systems and Computer Science (CSCS)*, IEEE, pp. 726–733, doi: 10.1109/CSCS.2019.00129.

Zdravevski, E., Lameski, P., Apanowicz, C. and Ślęzak, D. (2020), "From Big Data to business analytics: The case study of churn prediction", *Applied Soft Computing*, Vol. 90, p. 106164, doi: 10.1016/j.asoc.2020.106164.