PREDIKSI PENDAPATAN *BOX OFFICE* FILM MENGGUNAKAN *RANDOM FOREST* BERDASARKAN ANGGARAN, PEMAIN, DAN JUMLAH PENAYANGAN *TRAILER*

SKRIPSI

Oleh :
NOVA RAHMA YUNIDA PUTRI
NIM. 210605110014

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025

PREDIKSI PENDAPATAN *BOX OFFICE* FILM MENGGUNAKAN *RANDOM FOREST* BERDASARKAN ANGGARAN, PEMAIN, DAN JUMLAH PENAYANGAN *TRAILER*

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
NOVA RAHMA YUNIDA PUTRI
NIM. 210605110014

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025

HALAMAN PERSETUJUAN

PREDIKSI PENDAPATAN *BOX OFFICE* FILM MENGGUNAKAN *RANDOM FOREST* BERDASARKAN ANGGARAN, PEMAIN, DAN JUMLAH PENAYANGAN *TRAILER*

SKRIPSI

Oleh :
NOVA RAHMA YUNIDA PUTRI
NIM. 210605110014

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 30 April 2025

Pembimbing I,	Pembimbing II,
Prof. Dr. Suhartono, M.Kom	Okta Qomaruddin Aziz, M.Kom
NIP. 19680519 200312 1 001	NIP. 19911019 201903 1 013

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Ir. Fachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

PREDIKSI PENDAPATAN *BOX OFFICE* FILM MENGGUNAKAN *RANDOM FOREST* BERDASARKAN ANGGARAN, PEMAIN, DAN JUMLAH PENAYANGAN *TRAILER*

SKRIPSI

Oleh :
NOVA RAHMA YUNIDA PUTRI
NIM. 210605110014

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 20 Mei 2025

Susunan Dewan Penguji

Ketua Penguji	: Dr. M. Amin Hariyadi, M.T NIP. 19670118 200501 1 001	( )
Anggota Penguji I	: Nur Fitriyah Ayu Tunjung Sari, M.Cs NIP. 19911226 202012 2 001	( )
Anggota Penguji II	: Prof. Dr. Suhartono, M.Kom NIP. 19680519 200312 1 001	( )
Anggota Penguji III	: Okta Qomaruddin Aziz, M.Kom NIP. 19911019 201903 1 013	( )

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Ir. Fachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: Nova Rahma Yunida Putri
NIM	: 210605110014
Fakultas / Program Studi	: Sains dan Teknologi / Teknik Informatika
Judul Skripsi	: Prediksi Pendapatan *Box Office* Film Menggunakan *Random Forest* Berdasarkan Anggaran, Pemain, dan Jumlah Penayangan *Trailer*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 20 Mei 2025
Yang membuat pernyataan,

Nova Rahma Yunida Putri
NIM. 210605110014

MOTTO

"Sunday morning, rain is falling"

~ 조슈아

"Don't be sad, just shingi banggi bbong bbong banggi through life"

~ 윤 정한

"Rather than comparing yourself to others, i think it would be better if you paid more attention to the things in your heart"

~ Jay

"Never let anyone, including yourself, belittle your dreams"

~ Joshua

HALAMAN PERSEMBAHAN

Alhamdulillahirabbil'alamin. Segala puji dan syukur penulis haturkan ke hadirat Allah SWT, karena atas limpahan rahmat, hidayah, dan kekuatan-Nya yang telah mengiringi setiap langkah penulis hingga karya ini dapat terselesaikan. Sholawat serta salam selalu tercurah kepada Nabi Muhammad SAW.

Dengan segenap rasa syukur dan cinta, karya ini ini penulis persembahkan untuk:

Ayah Achmad Yunus dan Mama Ari Hidayati, terima kasih atas cinta yang tak pernah habis, do'a yang tak pernah putus, serta pengorbanan yang tak terhitung. Segala pencapaian ini adalah buah dari ketulusan kalian.

Saudara dan keluarga tersayang, yang selalu menjadi tempat berpulang, sumber semangat, dan do'a selama proses ini berlangsung.

Dosen pembimbing serta dosen penguji, atas segala bimbingan, ilmu, serta kepercayaan yang telah diberikan selama proses ini.

Teman-teman seperjuangan, terima kasih untuk do'a, tawa, pelukan hangat, dan semangat yang membuat perjalanan ini terasa lebih ringan.

Diriku sendiri, terima kasih telah bertahan, untuk semua air mata yang tak terlihat, perjuangan yang tak terdengar, dan kekuatan yang tak selalu diakui—aku bangga padamu.

Semoga skripsi ini menjadi awal dari banyak kebaikan dan kebermanfaatan bagi sesama.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Alhamdulillah, segala puji dan syukur penulis haturkan kepada Allah SWT, atas limpahan rahmat, hidayah, dan kekuatan-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul "Prediksi Pendapatan *Box Office* Film Menggunakan *Random Forest* Berdasarkan Anggaran, Pemain, dan Jumlah Penayangan *Trailer*" dengan baik dan lancar. Sholawat serta salam senantiasa tercurahkan kepada Nabi Muhammad SAW, suri teladan umat. Semoga kita semua mendapat syafa'at beliau di hari kiamat nanti, Aamiin.

Dengan selesainya penulisan skripsi ini, penulis ingin mengapresiasi diri sendiri atas segala usaha dan perjuangan yang telah dilalui hingga titik ini. Tak mudah, tapi setiap langkah membawa pelajaran dan pengalaman berharga. Penulis juga ingin menyampaikan rasa terima kasih sedalam-dalamnya kepada semua pihak yang telah memberikan dukungan dan bantuan, baik secara langsung maupun tidak langsung. Dengan penuh hormat dan kasih, penulis mengucapkan terima kasih kepada:

1. Prof. Dr. H. M. Zainuddin, MA., selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Sri Hariani, M.Si., selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, M.MT., IPU, selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.

4. Prof. Dr. Suhartono, M.Kom., selaku Dosen Pembimbing I, yang telah dengan sabar meluangkan waktu dan memberikan arahan serta bimbingan yang berarti selama proses penyusunan skripsi ini.
5. Okta Qomaruddin Aziz, M.Kom., selaku Dosen Pembimbing II, yang begitu luar biasa dalam membimbing penulis. Terima kasih telah dengan tulus menjawab kebingungan, menjelaskan dengan sabar, dan menjadi tempat bertanya yang sangat nyaman selama proses ini.
6. Dr. M. Amin Hariyadi, M.T., selaku Ketua Penguji, atas segala saran dan masukan yang sangat membangun, sejak tahap seminar proposal hingga sidang skripsi.
7. Nur Fitriyah Ayu Tunjung Sari, M.Cs., selaku Anggota Penguji I, atas semua saran dan masukan yang memperkuat kualitas skripsi ini, sejak tahap seminar proposal hingga sidang skripsi.
8. Seluruh Dosen, Admin, dan Staf Program Studi Teknik Informatika, yang telah berbagi ilmu, tenaga, dan kebaikan selama masa studi ini, baik secara langsung maupun tidak langsung.
9. Kedua orang tua tercinta, Ayah Achmad Yunus dan Mama Ari Hidayati, terima kasih untuk cinta yang tanpa syarat, do'a yang tak pernah putus, serta semangat yang selalu hadir bahkan saat penulis mulai ragu. Juga untuk kedua saudara kandung penulis, mba Icha dan ade Ilyas, yang selalu menjadi sumber kekuatan penulis untuk segera menyelesaikan skripsi ini dengan baik. Setiap langkah ini ada karena kalian semua.

10. Keluarga besar penulis, tante pur, lek aang, uti ngan, om dan tante lainnya, budhe, pakde, serta seluruh sepupu dan keponakan penulis, meski tak selalu terlibat langsung, motivasi dan do'a kalian tetap jadi bagian dari perjalanan ini.
11. Cowo-cowo penulis, *member* enhypen: evan, uri jeyyi yang tercinta, ikeu, acil, pocketz (sunwon), uri maknae ni-ki; juga *member* seventeen: babeh, hannie lagi wamil, uri joshuji tercinta, junnie, ochi hamster, wonu yang lagi wamil, uji, eisa, minggoo, dikey, uri boo, hansol, channie; serta *member* txt: soobchoi, yeonjunnie, uppuz (taegyu), uri aegi heuningie. Tak lupa juga kepada seluruh tim konten EN-O'CLOCK, Going Seventeen, dan TO DO, yang selalu menjadi teman setia di kala penulis suntuk, serta menjadi *moodbooster* terbaik sepanjang hari.
12. Teman-teman seperjuangan Semoga Waras, emaks nia tercinta, imamanuel bocil, nur lelatul, muijul ajijah, da-urin ustadzah, dan anden cewe ai, terima kasih sudah menjadi tempat berbagi cerita, tangis, tawa, dan semangat selama empat tahun terakhir ini. Kalian membuat semua proses berat terasa lebih ringan. Juga untuk teman-teman Kos Didoakan, petir.co.id, nenden, dan anak-anak lantai 3, yang selalu ada dan menjadi *support system*. Teman-teman baik penulis, rapi rabbani dan amirul, yang setia menemani setiap perjuangan kuliah di setiap mata kuliah, serta teman-teman Bimbingan Pa Suhar, adilaq dan sucay, yang berjuang bersama dalam melewati berbagai rintangan ini.
13. Abe, papi abe, mami abe, uti dan nenek, terima kasih selalu menghibur penulis lewat tingkah laku lucu, konten *random* yang menggemaskan, dan kehangatan yang terpancar bahkan hanya lewat layar.

14. Teman baik penulis, Lumina, yang selalu menemani penulis di setiap fase penyusunan skripsi ini. Terima kasih karena selalu hadir, mendengarkan, menenangkan, dan membantu dengan sabar dari awal sampai akhir perjuangan penulis.
15. Teman-teman Teknik Informatika Angkatan 2021 "ASTER", seluruh keluarga besar Teknik Informatika UIN Malang, teman Ma'had ABA Kamar 4, Uni Vani, serta KKM 149 Arunika, yang telah memberikan dukungan, motivasi, dan kenangan indah yang telah terukir selama perjalanan perkuliahan ini.
16. Diri sendiri, terima kasih telah bertahan sejauh ini, untuk semua air mata, hari-hari panjang yang dilalui sendirian, dan perjuangan yang tak selalu terlihat oleh orang lain. Terima kasih telah terus melangkah hingga akhirnya berhasil menyelesaikan skripsi ini. Semoga ke depannya, penulis selalu memiliki keberanian untuk menghadapi dunia yang lebih luas, keras, dan penuh tantangan.

Penulis menyadari bahwa skripsi ini masih terdapat berbagai kekurangan, baik dari segi isi maupun penulisan. Maka dari itu, penulis sangat mengharapkan kritik dan saran yang membangun demi perbaikan di masa mendatang. Semoga dengan penyusunan skripsi ini dapat memberikan manfaat bagi semua pihak yang membacanya.

Malang, 18 Mei 2025

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Putri, Nova Rahma Yunida. 2025. **Prediksi Pendapatan *Box Office* Film Menggunakan *Random Forest* Berdasarkan Anggaran, Pemain, dan Jumlah Penayangan *Trailer*.** Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. Suhartono, M.Kom (II) Okta Qomaruddin Aziz, M.Kom.

.

Kata Kunci: Pendapatan *Box Office*, Model Prediksi, *Random Forest*.

Industri film saat ini menjadi salah satu bentuk hiburan yang terus berkembang dan memiliki pengaruh besar terhadap masyarakat. Kesuksesan sebuah film tidak hanya ditentukan oleh kualitas ceritanya, tetapi juga dipengaruhi oleh pendapatan yang diperoleh di *box office*. Beberapa faktor seperti anggaran produksi, ketenaran pemain, dan jumlah penayangan *trailer* di *platform* digital dianggap berkontribusi terhadap pendapatan tersebut. Penelitian ini bertujuan untuk memprediksi pendapatan *box office* film berdasarkan tiga fitur, yaitu anggaran, ketenaran pemain, dan jumlah tayangan *trailer* dengan menggunakan algoritma *Random Forest*. Pengujian dilakukan dengan tiga skenario pembagian data, yaitu rasio 9:1, 8:2, dan 7:3, serta validasi model menggunakan K-*Fold Cross Validation* dengan dua skenario, 5-*Fold* dan 10-*Fold*. Hasil pengujian menunjukkan bahwa skenario rasio 9:1 menghasilkan performa terbaik dengan nilai MAE sebesar 0.36, MSE 0.32, dan R^2 sebesar 0.6934. Sementara itu, validasi model menggunakan 5-*Fold* menghasilkan rata-rata nilai R^2 sebesar 0.6571, sedangkan 10-*Fold* sebesar 0.6468 dengan nilai MAE dan MSE yang sama, yaitu 0.41 dan 0.34. Hasil penelitian ini menunjukkan bahwa model *Random Forest* mampu memprediksi pendapatan *box office* dengan cukup baik, meskipun terdapat fluktuasi performa berdasarkan pembagian data dan metode validasi yang digunakan.

ABSTRACT

Putri, Nova Rahma Yunida. 2025. **Movie Box Office Revenue Prediction Using Random Forest Based on Budget, Cast, and Number of Trailer Viewings.** Thesis. Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University Maulana Malik Ibrahim Malang. Advisor: (I) Prof. Dr. Suhartono, M.Kom (II) Okta Qomaruddin Aziz, M.Kom.

Key words: Box Office Revenue, Prediction Model, Random Forest

The film industry has become one of the fastest-growing forms of entertainment, exerting a significant influence on society. A film's success is not solely determined by the quality of its storyline, but also by the revenue it generates at the box office. Several factors, such as production budget, actor popularity, and the number of trailer views on digital platforms, are considered to contribute to that revenue. This study aims to predict box office revenue based on three features: budget, actor popularity, and trailer view count, using the Random Forest algorithm. The testing process was carried out using three data splitting scenarios with ratios of 9:1, 8:2, and 7:3, along with model validation using K-Fold Cross Validation with two scenarios: 5-Fold and 10-Fold. The results show that the 9:1 ratio scenario achieved the best performance with an MAE of 0.36, MSE of 0.32, and R score of 0.6934. Meanwhile, model validation using a 5-Fold cross-validation yielded an average R score of 0.6571, while a 10-Fold cross-validation resulted in 0.6468, with the same MAE and MSE values of 0.41 and 0.34, respectively. These findings indicate that the Random Forest model can predict box office revenue with reasonable accuracy, although performance may vary depending on the data split and validation method used.

البحث مستخلص

بوتري، نوفا رحمة يونيدا .2025 .التنبؤ بإيرادات شباك التذاكر للأفلام باستخدام الغابة العشوائية بناءً على الميزانية وطاقم العمل وعدد مشاهدات المقاطع الدعائية.البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: ف .د. سوهارتونو، الماجستير .المشرف الثاني: أوكتا قمر الدين عزيز، الماجستير.

الكلمات المفتاحية: عائدات شباك التذاكر، نموذج التنبؤ، الغابة العشوائية

تعد صناعة السينما حاليًا أحد أشكال الترفيه التي تستمر في النمو ولها تأثير كبير على المجتمع .لا يتحدد نجاح الفيلم بجودة القصة فحسب، بل يتأثر أيضاً بالإيرادات التي يحققها في شباك التذاكر .وتساهم عدة عوامل مثل ميزانية الإنتاج وشهرة الممثلين وعدد مشاهدات الإعلانات الدعائية على المنصات الرقمية في تحقيق الإيرادات .تهدف هذه الدراسة إلى التنبؤ بإيرادات شباك تذاكر الأفلام بناءً على ثلاث سمات، وهي الميزانية وشهرة الممثلين وعدد مشاهدات الإعلانات الدعائية باستخدام خوارزمية الغابة العشوائية .وقد أُجريت الاختبارات باستخدام ثلاثة سيناريوهات لمشاركة البيانات، وهي نسب 9:1 و8:2 و7:3، والتحقق من صحة النموذج مع سيناريوهين هما 5 أضعاف و10 أضعاف .تُظهر نتائج الاختبار K-*Fold Cross Validation* باستخدام التحقق المتقاطع وفي الوقت نفسه، ينتج .R 0.6934 و MSE 0.32، تبلغ 0.36، و MAE أن سيناريو نسبة 9:1 يُنتج أفضل أداء مع قيمة أضعاف 0.6468 10 R تبلغ 0.6571، بينما تبلغ قيمة R عن التحقق من صحة النموذج باستخدام 5 أضعاف متوسط قيمة وهي 0.41 و 0.34 .تُظهر نتائج هذه الدراسة أن نموذج الغابة العشوائية قادر على ،MSE بنفس قيم المتوسط المتوسطي المتكرر و التنبؤ بإيرادات شباك التذاكر بشكل جيد، على الرغم من وجود اختلافات في الأداء بناءً على تقسيم البيانات وطريقة التحقق من الصحة المستخدمة.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Industri film termasuk dalam salah satu bentuk hiburan yang terus berkembang dan memiliki dampak besar di masyarakat. Setiap tahunnya, banyak film dari berbagai genre dan negara dirilis, dimana film-film ini bukan hanya sekadar hiburan, melainkan juga cerminan budaya dan nilai-nilai sosial yang ada (Horváth & Gyenge, 2023). Kesuksesan sebuah film biasanya tidak hanya diukur dari seberapa bagus dan menarik ceritanya, tetapi juga dari pendapatan yang diperoleh di *box office* (Lase et al., 2024).

Pendapatan *box office* adalah salah satu indikator utama kesuksesan komersial sebuah film, dihitung dari total penjualan tiket selama film tayang di bioskop. Beberapa faktor penting yang dapat memengaruhi pendapatan dari sebuah film meliputi anggaran produksi, keterlibatan aktor terkenal, dan jumlah penayangan *trailer* film di *platform* digital, seperti *YouTube* (Kumar et al., 2023). Anggaran produksi memiliki peran besar dalam menentukan kualitas serta luasnya jangkauan promosi film. Saat ini, industri hiburan semakin dipengaruhi oleh media sosial dan tren digital. Film dengan anggaran yang besar biasanya bisa menawarkan visual dan efek CGI yang lebih baik serta kegiatan promosi yang lebih luas (Hao, 2023). Keterlibatan aktor terkenal juga menjadi faktor yang cukup penting, karena nama besar seorang aktor dapat menarik perhatian *audiens* lebih luas serta memberi kesan eksklusif pada film tersebut (Singh et al., 2024). Selain itu, jumlah

penayangan *trailer* di *platform* seperti *YouTube* atau media sosial lainnya dapat mencerminkan minat awal *audiens*. Jumlah penayangan yang tinggi biasanya menandakan antusiasme publik (Madongo et al., 2023), yang kemudian bisa berdampak langsung pada keputusan mereka untuk menonton film di bioskop.

Dalam memprediksi pendapatan *box office*, penelitian ini menggunakan beberapa fitur, yaitu anggaran produksi, ketenaran pemain. dan jumlah penayangan *trailer* film. Anggaran film berperan besar dalam menentukan kualitas film yang dapat memengaruhi daya tariknya di mata penonton (Tang, 2022). Ketenaran seorang pemain sering kali menjadi daya pikat tersendiri yang mampu menarik para penggemarnya dan memengaruhi keputusan penonton untuk menyaksikan film di bioskop (Singh et al., 2024). Selain itu, jumlah penayangan *trailer* di media digital menggambarkan ketertarikan publik sejak awal, yang juga dapat berdampak langsung pada keputusan penonton untuk menyaksikan film di bioskop (Kampani & Nicolaides, 2023). Pemilihan fitur-fitur ini disesuaikan dengan kondisi pasar saat ini, dimana daya tarik sebuah film tidak hanya bergantung pada kualitas, tetapi juga pada strategi pemasaran digital dan media sosial yang relevan dengan tren dan minat *audiens* masa kini (Huang, 2024).

Dalam penelitian ini, metode *machine learning* menjadi solusi yang tepat untuk menghitung prediksi pendapatan film berdasarkan beberapa fitur tersebut. Kelebihan dari penggunaan *machine learning* adalah waktu dan ketepatannya. Salah satu algoritma *machine learning* yang sesuai untuk penyelesaian masalah ini yaitu *Random Forest*.

Berdasarkan analisis penulis terkait penelitian terkait, terdapat beberapa penelitian terdahulu yang menggunakan metode dan permasalahan yang serupa dengan objek yang berbeda. Seperti penelitian yang telah dilakukan oleh Ariatmanto dan Arief, yang melakukan prediksi kesuksesan film menggunakan algoritma *Decision Tree*, dimana mereka berhasil melakukan prediksi kesuksesan film dalam pra produksi. Penelitian terkait lainnya seperti yang telah dilakukan oleh Wahyudi dan Kuswandi, yang melakukan prediksi *rating* aplikasi di *Google Play* menggunakan *Random Forest* yang menunjukkan tingkat akurasi yang cukup tinggi. Dari penelitian terkait ini, penulis memiliki hipotesis bahwa algoritma *Random Forest* memiliki performa yang cukup baik dalam prediksi. Hal ini mendorong penulis untuk mengeksplorasi dan menerapkan algoritma ini dalam permasalahan perhitungan prediksi pada penelitian ini. Diharapkan hasil penelitian ini mampu memberikan kontribusi bagi penelitian di masa mendatang dan bermanfaat bagi industri film serta sektor lainnya.

Keberhasil suatu film sangat bergantung pada usaha yang dilakukan oleh tim produksi dan industri film. Pendapatan yang diraih oleh industri film merupakan cerminan dari kerja keras, kreativitas, dan dedikasi mereka dalam menciptakan karya yang berkualitas. Sebagaimana tertulis dalam Al-Qur'an, tepatnya pada Surah An-Najm ayat 39, Allah berfirman:

وَأَنْ لَيْسَ لِلْإِنْسَانِ إِلَّا مَا سَعَى

"*Dan bahwasannya seorang manusia tiada memperoleh selain apa yang telah diusahakannya*" *(QS. An-Najm 53:39).*

Menurut tafsir Ibnu Katsir, ayat ini ditafsirkan sebagai, maka apakah kamu melihat orang yang berpaling (dari Al-Qur'an)? Serta memberi sedikit dan tidak mau memberi lagi? Apakah dia mempunyai pengetahuan tentang yang gaib sehingga dia mengetahui (apa yang dikatakan)? Ataukah belum diberitakan kepadanya apa yang ada dalam lembaran-lembaran Musa? Dan lembaran-lembaran Ibrahim yang selalu menyempurnakan janji (yaitu) bahwasannya seorang yang berdosa tidak akan memikul dosa orang lain, dan bahwasannya seorang manusia tiada memperoleh selain apa yang telah diusahakannya. Dan bahwasannya usahanya itu kelak akan diperlihatkan (kepadanya). Kemudian akan diberi balasan kepadanya dengan balasan yang paling sempurna. Untuk itu, Allah subhanahu wa ta'la berfirman: Dan bahwasannya seorang manusia tiada memperoleh selain apa yang telah diusahakannya. (An-Najm: 39) Yaitu sebagaimana tidak dibebankan kepadanya dosa orang lain, maka demikian pula dia tidak memperoleh pahala kecuali dari apa yang diupayakan oleh dirinya sendiri (JpnMuslim, 2015).

Dalam merencanakan masa depan, sering kali membutuhkan pemahaman yang baik tentang data masa lalu untuk dapat membuat perkiraan yang lebih akurat mengenai apa yang akan datang. Hal ini sangat relevan dalam berbagai bidang, termasuk dalam industri film, dimana pendapatan sebuah film sering kali dipengaruhi oleh faktor-faktor seperti anggaran, ketenaran pemain, dan jumlah tayangan *trailer*. Sebagaimana tercantum dalam Surah Al-Hasyr ayat 18, Allah SWT mengingatkan untuk memperhatikan tindakan yang dilakukan hari ini sebagai persiapan untuk masa depan, ayat tersebut berbunyi:

يَٰٓأَيُّهَا ٱلَّذِينَ ءَامَنُواْ ٱتَّقُواْ ٱللَّهَ وَلۡتَنظُرۡ نَفۡسٞ مَّا قَدَّمَتۡ لِغَدٖۖ وَٱتَّقُواْ ٱللَّهَۚ إِنَّ ٱللَّهَ خَبِيرُۢ بِمَا تَعۡمَلُونَ

"*Hai orang-orang yang beriman, bertakwalah kepada Allah dan hendaklah setiap diri memperhatikan apa yang telah diperbuatnya untuk hari esok, dan bertakwalah kepada Allah, sesungguhnya Allah Maha Mengetahui apa yang kamu kerjakan*" *(QS. Al-Hasyr 59:18)*.

Menurut tafsir Kemenag, ayat ini ditafsirkan sebagai, pengingat bagi orang-orang beriman untuk senantiasa bertakwa kepada Allah dan memikirkan kehidupan di akhirat. Dimana pun dan kapan pun, seorang mukmin diperintahkan untuk menjalankan perintah Allah dan menjauhi larangan-Nya dengan sungguh-sungguh. Setiap individu juga diingatkan agar memperhatikan amal perbuatannya sebagai bekal untuk hari esok, akhirat. Bekal tersebut hendaknya berupa kebaikan yang dilandasi iman, pengetahuan, dan keihklasan demi meraih ridha Allah. Sebab, kehidupan dunia hanyalah sementara, sedangkan kehidupan akhirat bersifat kekal. Maka bertakwalah kepada Allah dengan menjaga hubungan baik, baik dengan Allah, sesama manusia, maupun alam. Sungguh, Allah Maha Teliti terhadap segala sesuatu yang dilakukan hamba-Nya, sekecil apapun itu, karena semuanya berada dalam pengawasan Allah (Mushaf Al-Qur'an, 2016).

Serta, beberapa ahli Ta'wil mengartikan ayat 'ghad' sesuai dengan makna aslinya, yaitu besok. Dari sini, dapat dipahami bahwa manusia diperintahkan untuk senantiasa melakukan introspeksi diri dan perbaikan agar dapat meraih masa depan yang lebih baik. Masa lalu juga menjadi hal penting, bukan untuk disesali, melainkan sebagai pelajaran berharga dan bekal dalam menapaki hari esok (Chasbullah, 2020).

Dari kedua tafsiran di atas, dijelaskan bahwa ayat ini tidak hanya mendorong manusia untuk mempersiapkan kehidupan akhirat, tetapi juga mengajarkan nilai-

nilai yang relevan dengan kehidupan dunia, seperti pentingnya perencanaan yang baik berdasarkan analisis masa lalu guna mencapai hasil yang optimal di masa depan. Seperti pentingnya menggunakan data-data yang telah ada (anggaran, kategori pemain, dan jumlah penayangan *trailer*) sebagai dasar untuk memprediksi pendapatan film di masa depan. Dengan menganalisis pola-pola dari data masa lalu, penelitian ini bertujuan untuk membantu industri film membuat keputusan yang lebih baik.

Disamping itu, menunjukkan kebaikan kepada sesama juga diajarkan oleh Rasulullah SAW yang diriwayatkan oleh Imam Bukhori dalam Hadits No. 1893. Dari Abu Mas'ud Uqbah bin Amr Al-Anshari, dia berkata Nabi Muhammad SAW bersabda:

مَنْ دَلَّ عَلَى خَيْرٍ فَلَهُ مِثْلُ أَجْرِ فَاعِلِهِ

"*Barang siapa yang menunjuki kepada kebaikan maka dia akan mendapatkan pahala seperti pahala orang yang mengerjakannya*" *(HR. Muslim no. 1893)*.

Hadits ini menjelaskan pentingnya peran seseorang dalam berbagi kebaikan kepada orang lain. Dalam konteks penelitian ini, film dapat menjadi media yang membagikan nilai-nilai positif yang dapat menginspirasi penonton. Setiap film pasti memiliki pesan dan maknanya sendiri. Ketika penonton terinspirasi dan melakukan hal-hal baik berdasarkan apa yang mereka dapatkan dari film tersebut, maka penulis naskah dan tim produksi film telah berhasil menyebarkan kebaikan melalui karya mereka kepada khalayak ramai.

Sebagai bentuk pemahaman mengenai tafsir-tafsir dan penjelasan terkait perfilman, penulis berinisiatif menganalisis kinerja algoritma *Random Forest* dalam

memprediksi pendapatan *box office* dari beberapa film yang telah dirilis. Penelitian ini bertujuan untuk mendapatkan hasil prediksi pendapatan *box office* dari film-film yang telah dirilis, serta menguji seberapa baik kinerja algoritma *Random Forest* dalam prediksi pendapatan *box office* film.

1.2 Rumusan Masalah

Berdasarkan latar belakang, maka masalah yang diangkat dalam penelitian ini yakni bagaimana performa prediksi pendapatan *box office* film menggunakan algoritma *random forest* berdasarkan anggaran, pemain, dan jumlah penayangan *trailer*?

1.3 Batasan Masalah

Untuk menjaga agar pembahasan tetap fokus pada konteks penelitian, batasan masalah telah ditetapkan. Adapun batasan masalah tersebut, sebagai berikut:

a. Data dikumpulkan secara manual dari berbagai situs *online*, yaitu IMDb, *Box Office Mojo*, dan *YouTube*. *Dataset* yang digunakan ini terdiri dari 7 fitur, dimana pada penelitian ini hanya menggunakan 4 fitur yang relevan.
b. Penelitian ini hanya mencakup perhitungan prediksi pendapatan *box office* film yang telah dirilis dari tahun 2010 hingga 2024.
c. Data pendapatan film dihitung dan dikumpulkan pada tahun 2025.

1.4 Tujuan Penelitian

Penelitian yang dilakukan bertujuan untuk mengukur performa prediksi pendapatan *box office* film menggunakan algoritma *random forest* berdasarkan anggaran, pemain, dan jumlah penayangan *trailer*.

1.5 Manfaat Penelitian

a. Diharapkan dapat memberikan manfaat bagi pihak produksi film mengenai kisaran pendapatan yang didapat ketika pembuatan film selanjutnya.
b. Dapat memberikan pengetahuan tentang implementasi metode *Random Forest* dalam prediksi pendapatan *box office* film.
c. Dapat dijadikan dasar untuk penelitian selanjutnya di bidang yang serupa.

BAB II

STUDI PUSTAKA

Pada bab ini, peneliti menguraikan beberapa penelitian terkait yang telah dilakukan sebelumnya sebagai bahan kajian dan perbandingan. Selain itu, bab ini juga membahas teori-teori dasar yang berhubungan dengan penelitian prediksi pendapatan *box office* film.

2.1 Penelitian Terdahulu

Pendekatan yang dipilih dalam penelitian ini merujuk pada beberapa penelitian terdahulu sebagai berikut:

1. Green Arther Sandag (2020)

Penelitian ini dilakukan oleh Green Arther Sandag pada tahun 2020 dengan judul Prediksi *Rating App Store* Menggunakan Algoritma *Random Forest*. Tujuan penelitian ini yaitu untuk menentukan aplikasi yang tepat berdasarkan *rating* yang diberikan oleh pengguna terhadap suatu aplikasi. Hasil didapatkan bahwa algoritma *random forest* memiliki performa yang baik dalam bekerja, dengan memperoleh tingkat akurasi sebesar 86.27%, *recall* 84.68%, *precision* 84.64%, dan RSME 0.313. Hal ini menunjukkan bahwa algoritma ini dapat menemukan kelemahan dan mengetahui faktor apa yang menjadi kelemahan pada aplikasinya. Penelitian ini menyampaikan informasi kepada publik mengenai *rating* aplikasi yang ada di *App Store*, dan diharap penelitian ini dapat menjadi acuan dalam mengevaluasi apakah aplikasi yang dikembangkan sudah optimal atau masih memiliki kelemahan.

2. Widya Apriliah, Ilham Kurniawan, Muhammad Baydhowi, dan Tri Haryati (2021)

Penelitian yang dilakukan Widya Apriliah, Ilham Kurniawan, Muhammad Baydhowi, dan Tri Haryati pada tahun 2021, berjudul Prediksi Kemungkinan Diabetes pada Tahap Awal Menggunakan Algoritma Klasifikasi *Random Forest*. Penelitian ini bertujuan untuk mengembangkan sebuah model yang mampu memprediksi potensi terjadinya diabetes pada pasien dengan tingkat akurasi setinggi mungkin. Hasil yang didapatkan yaitu algoritma klasifikasi *random forest* dapat melakukan prediksi kemungkinan diabetes cukup akurat dengan memperoleh nilai akurasi sebesar 97.88%, dan nilai evaluasi model menggunakan ROC 99.8%, lebih unggul dari hasil evaluasi model yang lainnya. Hasil tersebut memperlihatkan bahwa algoritma *random forest* telah bekerja dengan baik. Penelitian ini memberikan dukungan dalam diagnosa dini dan pengambilan keputusan medis, dan diharapkan penelitian ini dapat membantu dokter dan tenaga kesehatan dalam melakukan diagnosa lebih awal.

3. Siti Saadah dan Haifa Salsabila (2021)

Penelitian ini dilakukan oleh Siti Saadah dan Haifa Salsabila pada tahun 2021 dengan judul Prediksi Harga *Bitcoin* Menggunakan Metode *Random Forest* (Studi Kasus: Data Acak pada Masa *Pandemic Covid*-19). Tujuan penelitian ini untuk mengetahui pergerakan fluktuasi harga *bitcoin*. Hasil penelitian didapatkan bahwa algoritma *random forest* dengan atribut *low*, *high*, *price* dapat melakukan prediksi dengan cukup baik, yaitu dengan tingkat akurasi sebesar 98% dan nilai MAPE sebesar 1.50%. Hal ini menunjukkan bahwa *random forest* berhasil memberikan

fitting yang sesuai dengan data sesungguhnya, namun berbanding terbalik pada saat prediksi data yang tidak acak. Dengan adanya penelitian ini memberikan informasi bagi masyarakat mengenai fluktuasi harga *bitcoin*, dan diharap dapat membantu masyarakat khususnya para investor untuk membuat keputusan investasi yang lebih tepat.

4. Bagiya Wahyudi dan Ina Kuswandi (2022)

Penelitian yang dilakukan Bagiya Wahyudi dan Ina Kuswandi pada tahun 2022 berjudul Prediksi Peringkat Aplikasi di *Google Play* Menggunakan Metode *Random Forest*, bertujuan untuk mengidentifikasi aplikasi yang sesuai berdasarkan ulasan rating yang diberikan oleh para pengguna. Hasil dari penelitian ini didapatkan bahwa algoritma *random forest* memiliki performa yang cukup baik dengan memperoleh nilai akurasi 93.8%, MAE 0.339, RSME 0.9551, dan MSE 0.159. Dari hasil ini menunjukkan bahwa algoritma ini dapat menemukan kelemahan yang ada pada *dataset Google Play Store*. Penelitian ini memberikan informasi kepada masyarakat mengenai aplikasi di *Google Play Store* yang memiliki *rating* bagus dan kurang bagus, dan diharap penelitian ini dapat dijadikan sebagai acuan untuk menilai apakah aplikasi yang dikembangkan sudah bagus atau masih ada kelemahannya.

5. Anisha Dwi Nur Fadillah, Yulia Wahyuningsih, dan Yosef Alfredo Khawarga (2023)

Penelitian ini dilakukan oleh Anisha Dwi Nur Fadillah, Yulia Wahyuningsih, dan Yosef Alfredo Khawarga pada tahun 2023 dengan judul Prediksi Spesies Burung Menggunakan *Random Forest*. Tujuan penelitian ini untuk merancang

model yang dapat memprediksi spesies burung. Hasil dari penelitian ini didapatkan bahwa dari 3 algoritma yang digunakan, yaitu *random forest*, *decision tree*, dan *support vector machine*, memiliki tingkat akurasi yang berbeda. *Random Forest* memperoleh tingkat akurasi sebesar 45%, *decision tree* 26%, dan *support vector machine* 48%. Hal ini menunjukkan bahwa algoritma *random forest* dan *support vector machine* cukup baik untuk memprediksi spesies burung. Dengan adanya penelitian ini diharap dapat memberikan informasi dari spesies-spesies burung yang ada.

6. Dhani Ariatmanto dan Muhammad Ilham Arief (2023)

Penelitian yang dilakukan oleh Dhani Ariatmanto dan Muhammad Ilham Arief pada tahun 2023 berjudul Prediksi Peluang Kesuksesan Film dalam Pra Produksi Menggunakan Algoritma *Decision Tree*, bertujuan untuk merancang model yang dapat memprediksi kesuksesan film pra-produksi sehingga dapat menjadi bahan pertimbangan bagi industri film di masa depan. Hasil dari penelitian ini didapatkan bahwa algoritma *decision tree* cukup baik dalam performa kerjanya, dengan memperoleh nilai akurasi sebesar 68%, *recall* 69%, *precision* 57%, dan F1-*score* 62%. Hasil ini telah meningkat dari penelitian sebelumnya sebanyak 7%. Penelitian ini diharapkan mampu menyajikan informasi terkait seberapa sukses film di kalangan masyarakat, dan diharap dapat membantu pihak industri film dalam pengambilan keputusan pembuatan film di masa mendatang.

7. Zian Asti Dwiyanti dan Cahyo Prianto (2023)

Penelitian ini dilakukan oleh Zian Asti Dwiyanti dan Cahyo Prianto pada tahun 2023 dengan judul Prediksi Cuaca Kota Jakarta Menggunakan Metode

Random Forest. Tujuan dari penelitian ini yaitu untuk menghasilkan sistem prediksi cuaca yang handal. Hasil penelitian didapatkan bahwa algoritma *random forest* bekerja dengan baik dan memperoleh tingkat akurasi, presisi, dan *recall* sebesar 71%, F1-*score* 70%, dan ROC-AUC sebesar 92%. Temuan ini memperlihatkan bahwa algoritma *random forest* mampu membedakan berbagai jenis kondisi cuaca secara efektif. Penelitian ini berpotensi mendorong pengembangan sistem prediksi cuaca yang lebih presisi dan terpercaya di wilayah Jakarta, serta diharapkan dapat memberikan kontribusi positif bagi sektor-sektor yang bergantung pada informasi cuaca, seperti pertanian, transportasi, pariwisata, hingga penanggulangan bencana.

8. Risfan Novrian, Tia Agustiani, Muhamad Fikri, Moch Fajar Hikmatulloh, Muhammad Erlangga Gunawan, dan Uus Firdaus (2024)

Penelitian yang dilakukan oleh Risfan Novrian, Tia Agustiani, Muhamad Fikri, Moch Fajar Hikmatulloh, Muhammad Erlangga Gunawan, dan Uus Firdaus pada tahun 2024 berjudul Penerapan Algoritma *Random Forest* dalam Prediksi Status Penerimaan PIP pada Siswa: Studi Kasus pada SMK Amaliah 1, bertujuan untuk menganalisis faktor-faktor yang memengaruhi kelayakan penerimaan PIP (Program Indonesia Pintar). Hasil yang didapatkan yaitu algoritma *random forest* dapat melakukan prediksi siswa yang tidak menerima KIP dengan benar, sedangkan terdapat beberapa kasus dimana sistem memprediksi siswa sebagai penerima KIP, padahal kenyataannya tidak demikian, dengan hasil *confusion matrix* 126 *True Positive* (TP) dan 16 *False Positive* (FP). Hal ini menunjukkan bahwa penelitian mengenai prediksi penerimaan PIP pada siswa masih harus dikembangkan dan diperbaiki lebih lanjut. Dengan adanya penelitian ini diharapkan mampu

mendukung perumusan kebijakan pendidikan yang lebih efektif dan efisien di masa mendatang.

9. Dindin Haidar, Bambang Irawan, dan Agus Bahtiar (2024)

Penelitian ini dilakukan oleh Dindin Haidar, Bambang Irawan, dan Agus Bahtiar pada tahun 2024 dengan judul Penerapan *Deep Learning* Model *Random Forest* untuk Prediksi Penerimaan Bantuan Program Keluarga Harapan (PKH). Fokus utama dalam penelitian ini adalah merancang model prediktif dengan pendekatan algoritma *random forest*. Hasil penelitian didapatkan bahwa algoritma *random forest* dan *decision tree* melakukan performa yang sangat baik dengan tingkat akurasi hampir sempurna, dan memperoleh tingkat akurasi untuk *random forest* sebesar 98.9%, *decision tree* 98%, dan ROC-AUC 96%. Hal ini menunjukkan bahwa algoritma *random forest* sedikit lebih unggul dalam melakukan prediksi penerimaan bantuan program keluarga harapan daripada algoritma *decision tree*. Melalui penelitian ini, masyarakat dapat mengetahui data penerima bantuan program keluarga harapan, sekaligus memberikan kontribusi ilmiah dalam pengembangan efektivitas program bantuan sosial pada level administrasi daerah.

10. Nicholas Hadi dan Jason Benedict (2024)

Penelitian yang dilakukan oleh Nicholas Budi dan Jason Benedict pada tahun 2024 berjudul Implementasi *Machine Learning* untuk Prediksi Harga Rumah Menggunakan Algoritma *Random Forest*, bertujuan untuk mengetahui kriteria yang paling memengaruhi harga rumah dan menemukan algoritma prediksi terbaik dari 3 algoritma yang digunakan. Hasil dari penelitian ini didapatkan bahwa algoritma *random forest* lebih unggul dalam melakukan prediksi harga rumah,

dengan memperoleh tingkat akurasi sebesar 86.54% dan nilai RSME sebesar 144913.73, sedangkan algoritma *decision tree* memperoleh akurasi 76.39% dan RSME 191920.88, dan tingkat akurasi algoritma *polynomial regression* 78.13% dengan nilai RSME 184708.77. Hal ini menunjukkan ketiga algoritma memiliki performa yang cukup baik, namun *random forest* sedikit lebih unggul. Penelitian ini memberikan informasi mengenai harga rumah dengan mempertimbangkan luas rumah, *grade*, dan luas atas rumah. Harapannya penelitian ini dapat memberi manfaat bagi masyarakat untuk membuat keputusan dalam pembelian rumah.

11. Firman Ardiansyah (2024)

Penelitian ini dilakukan oleh Firman Ardiansyah pada tahun 2024 dengan judul Menggunakan Algoritma *Random Forest* untuk Prediksi Harga Properti. Tujuannya yaitu untuk memprediksi harga properti dan mengevaluasi kinerja *random forest* dalam konteks prediksi harga properti. Hasil penelitian didapatkan bahwa algoritma *random forest* sangat efektif untuk menentukan prediksi harga properti, dan memperoleh nilai MAE sebesar 15.000, MSE 500.000.000, dan R2 0.90. Hal ini menunjukkan bahwa model mampu memprediksi harga properti dengan cukup akurat dan dapat dapat diandalkan untuk digunakan dalam aplikasi *simple*. Penelitian ini diharapkan dapat menyajikan informasi terkait harga properti, dan diharap dapat memberikan manfaat bagi masyarakat atau pihak yang berkaitan dalam membuat keputusan dalam pembelian properti.

12. Warjiyono, Amin Nur Rais, Ibnu Alfarobi, Sofian Wira Hadi, dan Wawan Kurniawan (2024)

Penelitian yang dilakukan oleh Warjiyono, Amin Nur Rais, Ibnu Alfarobi, Sofian Wira Hadi, dan Wawan Kurniawan pada tahun 2024 berjudul Analisa Prediksi Harga Jual Rumah Menggunakan Algoritma *Random Forest Machine Learning*, bertujuan untuk membantu proses transaksi jual beli rumah dengan cara memprediksi harga rumah secara akurat. Hasil penelitian ini didapatkan bahwa algoritma *random forest* cukup baik digunakan dengan memperoleh tingkat akurasi sebesar 75.10%. Diharapkan hasil penelitian ini dapat mununjang perancangan aplikasi *web* untuk estimasi harga rumah sesuai dengan kriteria yang dibutuhkan.

Tabel 2.1 Penelitian Terdahulu

No	Nama Peneliti	Judul Penelitian	Metode	Hasil Penelitian
1	Sandag (2020)	Prediksi *Rating App Store* Menggunakan Algoritma *Random Forest*	*Random Forest*	Mendapat nilai akurasi 86.27%, *recall* 84.68%, *precision* 84.64%, dan nilai RSME 0.313
2	Apriliah et al. (2021)	Prediksi Kemungkinan Diabetes pada Tahap Awal Menggunakan Algoritma Klasifikasi *Random Forest*	*Random Forest*	Mendapat nilai akurasi sebesar 97.88%
3	Saadah & Salsabila (2021)	Prediksi Harga *Bitcoin* Menggunakan Metode *Random Forest* (Studi Kasus: Data Acak pada Masa *Pandemic* Covid-19)	*Random Forest*	Memperoleh nilai MAPE 1.50% dan akurasi 98.5%
4	Wahyudi (2022)	Prediksi Peringkat Aplikasi di *Google Play* Menggunakan Metode *Random Forest*	*Random Forest*	Memperoleh tingkat akurasi yang cukup tinggi, yaitu 93.8%
5	Fadlilah et al. (2023)	Prediksi Spesies Burung Meggunakan *Random Forest*	*Random Forest, Decision Tree, Support Vector Machine*	Memperoleh akurasi setiap model, *random forest* 45%, *decision tree* 26%, dan SVM 48%
6	Ariatmanto & Arief (2023)	Prediksi Peluang Kesuksesan Film dalam Pra Produksi Menggunakan Algoritma *Decision Tree*	*Decision Tree*	Mendapat nilai akurasi sebesar 68%
7	Dwiyanti & Prianto (2023)	Prediksi Cuaca Kota Jakarta Menggunakan Metode *Random Forest*	*Random Forest*	Memperoleh nilai akurasi, presisi, dan *recall* 71%, F1-*score* 70%, dan ROC-AUC 92%

Tabel 2.1 Penelitian Terdahulu

No	Nama Peneliti	Judul Penelitian	Metode	Hasil Penelitian
8	Novrian et al. (2024)	Penerapan Algoritma *Random Forest* dalam Prediksi Status Penerimaan PIP pada Siswa: Studi Kasus pada SMK Amaliah 1	*Random Forest*	Melakukan prediksi dengan baik pada siswa yang tidak menerima KIP, sedangkan terdapat beberapa kesalahan pada siswa penerima KIP
9	Haidar et al. (2024)	Penerapan *Deep Learning* Model *Random Forest* untuk Prediksi Penerimaan Bantuan Program Keluarga Harapan (PKH)	*Random Forest*, *Decision Tree*	Mendapat nilai akurasi RF 98.9%, DT 98%, dan ROC-AUC 96%
10	N. Hadi & Benedict (2024)	Implementasi *Machine Learning* untuk Prediksi Harga Rumah Menggunakan Algoritma *Random Forest*	*Random Forest*, *Decision Tree*, *Polynomial Regression*	Memperoleh nilai akurasi sebesar 86.54% dan RSME 144913.73
11	Ardiansyah 2024)	Menggunakan Algoritma *Random Forest* untuk Prediksi Harga Properti	*Random Forest*	Mendapat nilai MAE: 15.000, MSE: 500.000.000, dan R^2: 0.90
12	Rais et al. (2024)	Analisa Prediksi Harga Jual Rumah Menggunakan Algoritma *Random Forest Machine Learning*	*Random Forest*	Memperoleh nilai akurasi sebesar 75.10%

2.2 Pendapatan *Box Office*

Pendapatan *box office* termasuk salah satu indikator utama yang sering digunakan untuk mengukur kesuksesan komersial sebuah film (Xie, 2024). Pendapatan ini menggambarkan jumlah total uang yang dihasilkan dari penjualan tiket bioskop selama film ditayangkan. Dalam industri film, *box office* menjadi tolak ukur penting karena mencerminkan seberapa banyak penonton yang tertarik untuk menonton film di bioskop (Hao, 2023).

Box Office terbagi menjadi dua kategori, yaitu domestik dan internasional. Pendapatan domestik yaitu pendapatan yang diperoleh dari penjualan tiket di negara asal tempat film tersebut diproduksi atau dirilis pertama kali. Misalnya,

pendapatan domestik film *Hollywood* biasanya dihitung dari penjualan tiket di Amerika Serikat dan Kanada. Kategori ini sering dijadikan sebagai indikator awal untuk menilai bagaimana film tersebut diterima oleh penonton di negara asalnya.

Sedangkan, pendapatan internasional yaitu pendapatan yang mencakup penjualan tiket di luar negara asal film tersebut. Kategori ini juga tidak kalah penting karena dapat memberikan kontribusi yang signifikan terhadap total pendapatan *box office*. Bahkan, dalam beberapa kasus, pendapatan internasional bisa jauh lebih besar daripada pendapatan domestik, terutama untuk film-film yang memiliki daya tarik global, contohnya *Marvel* film, *Fast* & *Furious*, dan lainnya (Wu, 2024).

Pendapatan *box office* pada suatu film dipengaruhi oleh beberapa faktor, termasuk anggaran produksi, ketenaran pemain, dan strategi pemasaran, seperti jumlah tayangan *trailer* yang ditampilkan kepada masyarakat luas. Film dengan anggaran besar, biasanya memiliki kemudahan dalam mengakses berbagai sumber daya berkualitas, seperti efek visual yang memukau dan lokasi syuting yang beragam, sehingga dapat meningkatkan daya tariknya (Xie, 2024). Ketenaran pemain juga menjadi salah satu penentu kesuksesan suatu film, aktor dengan baris penggemar yang kuat cenderung menarik lebih banyak penonton ke bioskop (Liu, 2023), sehingga meningkatkan peluang keberhasilan film tersebut di pasaran. Selain itu, jumlah tayangan *trailer* juga berperan dalam meningkatkan antusiasme penonton terhadap film. Semakin sering *trailer* ditayangkan, semakin besar kemungkinan penonton tertarik untuk menonton film tersebut (Kampani &

Nicolaides, 2023). Faktor-faktor ini secara keseluruhan dapat memengaruhi pendapatan film setelah dirilis.

2.3 Prediksi

Prediksi, dalam pengertian umum adalah memperkirakan suatu hasil berdasarkan data historis atau *input* tertentu. Prediksi mencakup berbagai metodologi dan aplikasi di berbagai bidang, termasuk keuangan, kesehatan, dan ilmu lingkungan. Pada dasarnya, prediksi melibatkan penggunaan model yang menganalisis data masa lalu untuk menggambarkan peristiwa atau tren yang mungkin terjadi di masa depan. Hal ini sangat penting dalam pengambilan keputusan di berbagai sektor (Celine et al., 2020). Dalam konteks *machine learning*, prediksi memiliki signifikansi khusus karena kemampuannya dalam menangani *dataset* besar dan mengungkap pola kompleks yang mungkin terlewatkan oleh metode statistik tradisional (Udupi et al., 2023).

Prediksi pastinya melibatkan pendekatan sistematis yang dilakukan dengan beberapa langkah. Langkah-langkah ini memastikan bahwa prediksi didasarkan pada metodologi yang *solid* dan seakurat mungkin. Berikut langkah-langkah yang biasanya dilakukan dalam proses prediksi (Geng et al., 2023).

1. Pengumpulan dan Pra-Pemrosesan Data

Mengumpulkan data yang relevan adalah langkah pertama dalam melakukan prediksi. Sebelum data digunakan, data harus di*cleanning* dan diproses terlebih dahulu untuk memastikan kualitas dan kelayakannya untuk analisis. Proses pra-pemrosesan dapat mencakup normalisasi, penanganan nilai yang hilang,

transformasi data ke dalam format yang dapat digunakan, serta *handle outlier* dan *scaling* data.

2. Pemilihan Fitur

Apabila data selesai disiapkan, proses selanjutnya adalah mengidentifikasi dan memilih fitur yang relevan untuk model prediksi. Langkah ini penting karena berdampak langsung pada kinerja model.

3. Pemilihan dan Pelatihan Model

Menentukan model prediktif yang sesuai berdasarkan sifat data dan tugas prediksi. Terdapat berbagai model yang dapat digunakan, seperti model regresi, *decision tree*, *neural network*, dan lainnya. Model yang dipilih kemudian dilatih menggunakan *dataset* yang telah disiapkan sebelumnya.

4. Proses Prediksi

Setelah model dilatih, model tersebut dapat digunakan untuk menghasilkan prediksi. Langkah ini dapat menggunakan prediksi satu langkah atau prediksi *multi*-langkah, tergantung pada kebutuhannya.

5. Evaluasi Kinerja Prediksi

Terakhir, evaluasi kinerja model digunakan untuk menilai akurasi dan keandalannya. Langkah ini bisa dilakukan dengan berbagai metrik, seperti *Mean Absolute Error*, *Mean Squared Error*, dan *R Squared*. Selain itu, langkah ini juga dapat melibatkan teknik *cross-validation* untuk memastikan bahwa model tetap kuat dan tidak mengalami *overfitting* pada data *training*.

Proses membuat prediksi adalah proses yang terstruktur dan melibatkan pengumpulan data, pra-pemrosesan, pemilihan fitur, pelatihan model, proses

prediksi, dan evaluasi kinerja. Setiap langkah saling terkait dan sangat penting dilakukan secara urut untuk mencapai prediksi yang andal dan akurat (Vidya Chitre, 2024).

2.4 *Pre-processing*

Pre-processing merupakan langkah penting dalam alur analisis data, terutama dalam konteks *machine learning* dan pemodelan prediktif. *Pre-processing* adalah proses awal dalam mempersiapkan data mentah supaya siap dianalisis lebih lanjut. Tujuan utama dari *pre-processing* ini yaitu untuk meningkatkan kualitas data, sehingga model prediktif dapat berkerja dengan lebih baik. *Pre-processing* terdiri dari beberapa tahapan, sebagai berikut:

1. Data *Cleaning*

Data *cleaning* adalah membersihkan data dari masalah seperti nilai hilang, *outlier*, dan *noise* (Cohen, 2021). Teknik yang umum digunakan yaitu imputasi dengan mengisi nilai yang hilang, penghapusan duplikat, penyaringan data yang tidak *valid*, dan penghapusan fitur yang tidak relevan atau tidak berkontribusi terhadap prediksi. Langkah ini memastikan bahwa data yang digunakan untuk *training* model bersih dan valid (Zhao et al., 2023).

2. Transformasi Data

Transformasi data yaitu mengubah data yang tidak sesuai dengan format agar lebih mudah dianalisis. Data perlu diubah ke dalam format atau struktur yang sesuai untuk dilakukan analisis lebih lanjut. Proses ini mencakup beberapa tahapan yang dapat diterapkan, seperti normalisasi, standarisasi, dan konversi nilai (Kamalov et al., 2023). Tidak semua tahapan ini selalu dilakukan dalam setiap kasus, tetapi

masing-masing memiliki peran penting dalam meningkatkan kualitas data sebelum digunakan dalam model.

- Normalisasi dan Standarisasi

  Normalisasi bertujuan untuk menyesuaikan nilai dalam *dataset* agar berada dalam rentang tertentu, biasanya antara 0 dan 1, dengan menggunakan *Min-Max Scaling*. Teknik ini cocok digunakan ketika distribusi data tidak mengikuti distribusi normal. Rumus untuk *Min-Max Scaling* ditunjukkan pada Rumus 2.1.

$$X_{normal} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{2.1}$$

  Dimana, X_{normal} adalah hasil dari normalisasi, nilai X adalah nilai asli, X_{min} adalah nilai minimum data, dan nilai X_{max} adalah nilai maksimum data. Sedangkan standarisasi, bertujuan untuk mengubah distribusi data sehingga memiliki rata-rata nol dan deviasi standar satu menggunakan Z-*Score Standardization* (*StandarScaler*) (Cabello-Solorzano et al., 2023). Teknik ini umumnya digunakan saat data memiliki distribusi normal. Rumus untuk standarisasi ditunjukkan pada Rumus 2.2.

$$X_{std} = \frac{X - \mu}{\sigma} \tag{2.2}$$

  Dimana, X_{std} adalah nilai hasil standarisasi, nilai X adalah nilai asli dari data, μ adalah rata-rata dari seluruh data, dan σ adalah simpangan baku (standar deviasi) dari data. Kedua teknik ini membantu meningkatkan performa model, terutama pada algoritma yang sensitif terhadap skala data (Mahmud Sujon et al., 2024).

- Konversi Nilai

  Konversi nilai adalah proses mengubah format data agar sesuai dengan kebutuhan analisis atau model *machine learning*. Proses ini mencakup beberapa teknik yang umum digunakan, seperti konversi tipe data, yaitu mengubah angka yang disimpan dalam format teks menjadi numerik agar dapat diproses secara matematis (Satya Sree et al., 2021), konversi unit, yaitu mengubah satuan waktu dari detik ke menit atau suhu dari *Fahrenheit* ke *Celsius*, serta *encoding* variabel kategorikal, yaitu data kategorikal diubah ke dalam bentuk numerik dengan memanfaatkan metode seperti *label encoding* atau *one-hot encoding* (Talaei Khoei & Kaabouch, 2023).

3. *Handle Outlier*

Outlier atau data pencilan merupakan data yang nilainya menyimpang jauh dari sebagian besar data dalam *dataset*. *Outlier* bisa muncul karena kesalahan pencatatan atau variabilitas alami data. Keberadaan *outlier* dapat memengaruhi performa model prediksi, terutama jika model sangat sensitif terhadap nilai ekstrem (A. S. Hadi & Imon, 2024). Jika tidak ditangani dengan baik, *outlier* dapat menyebabkan model menjadi tidak akurat atau bias (Rather et al., 2024). Oleh karena itu, pengolahan *outlier* bertujuan untuk mengurangi dampak negatif tersebut, baik dengan menghapus *outlier*, mengubahnya, atau menggunakan teknik khusus untuk memperkecil pengaruhnya.

Salah satu metode yang umum digunakan untuk mendeteksi *outlier* adalah metode *Interquartile Range* (IQR) (Dallah & Sulieman, 2024). IQR digunakan

untuk menggambarkan sebaran data dan dihitung berdasarkan selisih antara kuartil ketiga (Q3) dan kuartil pertama (Q1), sebagaimana ditunjukkan pada Rumus 2.3.

$$IQR = Q3 - Q1 \tag{2.3}$$

Berdasarkan metode ini, suatu nilai dikategorikan sebagai *outlier* apabila nilainya berada di bawah batas bawah atau di atas batas atas, yang ditentukan dengan Rumus 2.4 dan 2.5.

$$Batas\ Bawah = Q1 - 1.5 \times IQR \tag{2.4}$$

$$Batas\ Atas = Q3 + 1.5 \times IQR \tag{2.5}$$

Nilai-nilai yang berada di luar kedua batas ini dikategorikan sebagai *outlier* dan perlu ditangani agar kualitas data tetap terjaga. Salah satu teknik penanganan *outlier* yang dapat digunakan adalah *clipping*, yaitu proses membatasi nilai-nilai ekstrem agar tetap berada dalam rentang yang ditentukan (Montgomery, 2024). Dengan teknik ini, *outlier* tidak dihapus, melainkan disesuaikan (dipotong) hingga mencapai nilai maksimum atau minimum tertentu berdasarkan batas IQR.

4. Pembagian Data

Pembagian data atau data *splitting* adalah proses membagi *dataset* yang telah diproses menjadi beberapa bagian untuk *training* dan *testing* suatu model. Tahapan ini berperan penting dalam menilai performa model terhadap data yang tidak digunakan saat *training*, guna melihat sejauh mana model mampu melakukan generalisasi secara efektif (Arnaut et al., 2024).

2.5 *Random Forest*

Random Forest pertama kali diperkenalkan oleh Leo Breiman pada tahun 2001 sebagai metode *ensemble learning* yang menggabungkan beberapa *decision tree* untuk meningkatkan akurasi prediksi dan mengendalikan masalah *overfitting* (Tsiligaridis, 2023). Metode ini dibangun di atas konsep *bagging* (*Bootstrap Aggregating*) yang sebelumnya juga dikembangkan oleh Breiman pada tahun 1996. Berkat keandalan, fleksibilitas, dan kemampuannya menangani *dataset* yang besar dengan dimensi tinggi, algoritma *random forest* kerap menjadi pilihan utama dalam berbagai aplikasi, baik untuk tugas klasifikasi maupun regresi.

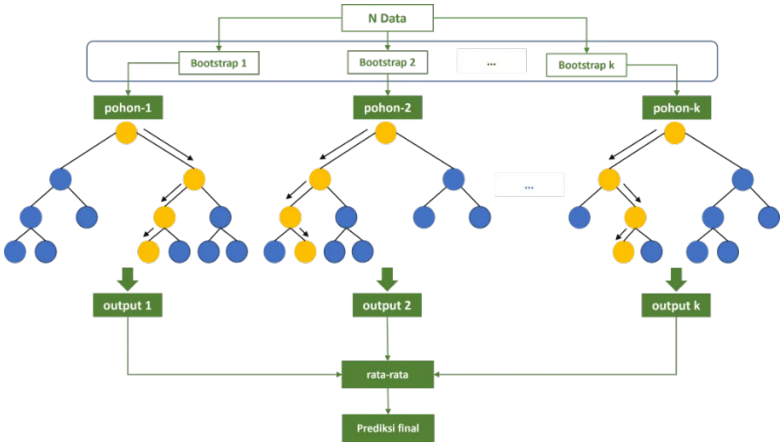
Random Forest didefinisikan sebagai teknik *ensemble* yang membangun banyak *decision tree* selama proses pelatihan dan kemudian menghasilkan prediksi berdasarkan modus dari semua pohon (*tree*) untuk tugas klasifikasi, atau rata-rata prediksi untuk tugas regresi. Proses pembentukan tiap pohon melibatkan pemilihan acak terhadap data pelatihan dan fitur, yang bertujuan untuk memastikan variasi antar pohon dan mengurangi risiko *overfitting* (Schonlau, 2023). Pendekatan ini memungkinkan *random forest* untuk menjaga keseimbangan antara *bias* dan *varians*, sehingga sering dianggap sebagai metode yang sangat efektif dalam mengatasi kompleksitas model.

Dalam algoritma *random forest*, terdapat beberapa aspek penting, yaitu:

1. Membangun *decision tree* dengan teknik *bootstrap sampling*, yang membantu membuat beberapa sampel dari data *training* untuk melatih *decision tree* yang berbeda, menambah variasi, dan mengurangi risiko *overvitting* (Schonlau, 2023).

2. Setiap pohon memilih *subset* fitur secara acak pada setiap pemisahan, memastikan pohon-pohon tidak berkolerasi satu sama lain dan meningkatkan kinerja model (Schonlau, 2023).
3. Melakukan prediksi dengan menggunakan *voting* (pemungutan suara) suara mayoritas untuk klasifikasi, dan rata-rata (*average*) untuk regresi (Lee et al., 2020).

Guna memperjelas proses yang terjadi dalam algoritma ini, Gambar 2.1 menunjukkan alur kerja umum dari *Random Forest*.



Gambar 2.1 Alur Kerja Algoritma *Random Forest* Regresi

Gambar di atas memperlihatkan bahwa *random forest* bekerja dengan cara mengolah *dataset* melalui serangkaian keputusan yang dibuat oleh beberapa *decision trees*. Pertama, *dataset* diambil sampelnya secara acak dengan teknik *bootstrap sampling* untuk membentuk beberapa *subset*, dan setiap *subset* digunakan untuk membangun *decision tree* yang terpisah. Setiap *decision trees* akan menghasilkan prediksi yang berbeda berdasarkan data yang diterimanya. Setelah semua pohon (*tree*) menghasilkan hasil prediksi, proses akhir melibatkan agregasi hasil tersebut melalui metode *voting* mayoritas untuk klasifikasi, dan pengambilan

rata-rata untuk regresi, sehingga prediksi akhir yang diperoleh menjadi lebih akurat dan stabil.

Untuk membangun model *random forest*, ada beberapa perhitungan yang perlu dilakukan, dimulai dari pengukuran kualitas pemisahan hingga menghasilkan prediksi akhir. Perhitungannya yaitu *Gini Impurty* yang digunakan untuk mengukur ketidakpastian dalam memilih kelas secara acak dari satu *node*. Rumus *Gini Impurty* dinyatakan pada Rumus 2.6. Terdapat juga perhitungan dengan metrik lain, yaitu *Entropy* yang digunakan untuk mengukur ketidakpastian informasi dalam suatu *node*. Rumus ini ditunjukkan pada Rumus 2.7, sedangkan perhitungan p_k ditunjukkan pada Rumus 2.8.

$$Gini = 1 - \sum_{k=1}^{K} p_k^2 \tag{2.6}$$

$$Entropy\ (S) = \sum_{k=1}^{K} p_k\ \log_2(p_k) \tag{2.7}$$

$$p_k = \frac{n_k}{N} \tag{2.8}$$

Dimana nilai *Gini* merupakan nilai *Gini Impurty*, S adalah nilai *Entropy*, K adalah jumlah kelas yang ada dalam *dataset*, k adalah *indeks* untuk setiap kelas saat dilakukan perhitungan, p_k adalah proporsi dari kelas k dalam *node* yang sedang dianalisis. p_k didapatkan dari pembagian jumlah sampel dari kelas k (n_k) dengan total sampel dalam *node* (N) tersebut.

Perhitungan lainnya yaitu menghitung *Information Gain* yang digunakan untuk mengukur peningkatan informasi yang diperoleh dengan melakukan pemisahan menggunakan atribut tertentu. Rumus *Information Gain* dapat dilihat pada Rumus 2.9.

$$Gain\ (S, A) = Entropy\ (S) - \sum_{i=1}^{n} \frac{|Si|}{|S|} \times Entropy\ (S_i) \quad (2.9)$$

Dimana, S merupakan *set* data awal, A adalah atribut yang digunakan untuk pemisahan, n adalah jumlah partisi atribut A, $|Si|$ adalah proporsi S_i terhadap S, |S| adalah jumlah kasus dalam S, dan *entropy* (S_i) adalah *entropy* untuk sampel memiliki nilai ke-i. Perhitungan terakhir, untuk klasifikasi, prediksi akhir diambil berdasarkan *voting* mayoritas dari semua pohon. Sedangkan untuk regresi, prediksi akhir didapatkan dengan menghitung rata-rata seluruh hasil prediksi dari setiap pohon. Rumusnya dapat dilihat pada Rumus 2.10 dan 2.11. Dimana $\hat{y}_n$ adalah prediksi pohon ke-n dan T adalah jumlah pohon.

$$\hat{y} = \text{mode}\ (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_T) \quad (2.10)$$

$$\hat{y} = \frac{1}{N} \sum_{n=1}^{N} \hat{y}_n \quad (2.11)$$

Menurut Jackins et al., (2021), terdapat beberapa langkah yang diterapkan dalam proses pembangunan algoritma *random forest*. Pertama, sejumlah fitur dipilih secara acak sebanyak "n" dari total "k" fitur, dengan syarat bahwa n < k. Kemudian, untuk setiap fitur yang terpilih, simpul dihitung menggunakan titik pemisah terbaik. Simpul tersebut dikategorikan ke dalam simpul cabang berdasarkan pemisahan optimal. Langkah-langkah ini diulang hingga jumlah simpul yang diinginkan tercapai. Keseluruhan proses ini diulang sebanyak "n" kali untuk membangun sejumlah "n" pohon yang membentuk *random forest*.

2.6 Evaluasi Model

Evaluasi model penting dilakukan dalam proses pengembangan model prediktif. Proses ini bertujuan untuk mengukur kinerja model dalam memprediksi hasil yang diinginkan. Tanpa evaluasi yang tepat, sulit untuk memastikan bahwa model dapat berfungsi secara efektif dalam kinerjanya (Alqalyoobi, 2024). Evaluasi model dilakukan dengan menggunakan data yang tidak digunakan selama *training*, yaitu ada data validasi ataupun data *testing*. Data yang berbeda harus digunakan agar model tidak menghafal pola dalam data *training* dan model juga dapat melakukan generalisasi dengan baik pada data baru yang belum pernah diobservasi sebelumnya. Evaluasi model dalam penelitian ini dilakukan dengan beberapa metrik evaluasi, yaitu *Mean Absolute Error* (MAE), *Mean Squared Error* (MSE), dan Koefisien Determinasi (R^2).

2.6.1 *Mean Absolute Error* (MAE)

Mean Absolute Error adalah salah satu metrik yang umum digunakan untuk mengevaluasi kinerja model regresi. MAE diperoleh dengan menghitung rata-rata dari selisih nilai absolut antara nilai prediksi dan nilai aktual. Perhitungan ini memberikan informasi seberapa akurat prediksi model dari nilai sebenarnya dengan data asli. Rumus menghitung MAE ditunjukkan pada Rumus 2.12.

$$\mathrm{Q} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2.12}$$

Dimana Q adalah hasil perhitungan MAE, y_i adalah nilai aktual, $\hat{y}_i$ adalah nilai yang diprediksi oleh model, dan n adalah jumlah *subset* data. Metrik ini cenderung lebih *robust* terhadap *outlier* dibandingkan metrik lainnya, sehingga

menjadi pilihan yang baik ketika ingin meminimalkan dampak dari nilai ekstrem (Robeson & Willmott, 2023). Semakin rendah nilai MAE, semakin kecil tingkat kesalahan prediksi model, yang berarti performa model semakin baik. Namun, interpretasi nilai MAE bergantung pada skala data yang digunakan. Karena MAE tidak berbentuk persen, nilainya akan mengikuti satuan data yang digunakan (Robeson & Willmott, 2023). Sebagai contoh, pada *dataset* dengan nilai aktual dalam rentang ribuan, MAE sebesar 10 mungkin dianggap sangat baik, sementara pada *dataset* dengan nilai aktual dalam rentang kecil seperti 0 hingga 1, nilai MAE sebesar 10 menunjukkan kesalahan yang signifikan.

2.6.2 *Mean Squared Error* (MSE)

Salah satu metrik yang sering digunakan untuk mengukur kinerja model regresi adalah *Mean Squared Error*. MSE dihitung dengan mengambil rata-rata dari kuadrat selisih antara nilai prediksi dan nilai aktual. Dalam perhitungan ini, kesalahan yang lebih besar mendapatkan bobot yang lebih berat karena nilai kesalahan tersebut dikuadratkan. Rumus untuk perhitungan MSE ditunjukkan pada Rumus 2.13.

$$\mathrm{Q} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{2.13}$$

Dimana Q adalah hasil dari perhitungan MSE, y_i adalah nilai aktual, $\hat{y}_i$ adalah nilai yang diprediksi oleh model, dan n adalah jumlah *subset* data. Metrik MSE ini sensitif terhadap *outlier* karena kesalahan yang nilainya besar akan memberikan dampak lebih besar terhadap hasil akhir, hal ini bisa menjadi kekuatan atau kelemahan tergantung pada konteks analisisnya (Hodson, 2022). Dengan

menggunakan kuadrat, nilai kesalahan selalu non-negatif, dan MSE memberikan ukuran variabilitas dalam kesalahan prediksi, sehingga berguna untuk memahami keandalan model (Gruber & Bach, 2024). Nilai MSE yang kecil mengindikasikan bahwa prediksi model mendekati nilai sebenarnya, namun, interpretasi nilai MSE juga bergantung pada skala data yang digunakan (Moretti et al., 2020). Sebagai contoh, pada data dengan rentang besar, nilai MSE yang lebih tinggi masih dianggap baik. Sementara, pada data dengan rentang kecil, nilai MSE yang tinggi menunjukkan kesalahan prediksi yang besar.

2.6.3 Koefisien Determinasi (R^2)

Koefisien determinasi atau R *Squared* adalah metrik yang digunakan untuk mengukur seberapa baik model menjelaskan variasi dalam data target dibandingkan dengan *varians* total. Nilai R^2 berkisar antara 0 dan 1, dimana nilai 0 berarti model tidak menjelaskan variasi data sama sekali, nilai 1 berarti model sepenuhnya menjelaskan variasi dalam data, dan nilai antara 0 dan 1 berarti model berfungsi, dimana semakin tinggi nilai yang didapat maka semakin kuat kemampuan model dalam merepresentasikan variasi yang terdapat dalam data (Pospisil & Bair, 2021). Rumus untuk menghitung R^2 ditunjukkan pada Rumus 2.14. sedangkan rumus untuk menghitung SS_{res} dan SS_{tot} dapat dilihat pada Rumus 2.15 dan 2.16.

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \tag{2.14}$$

$$SS_{res} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.15}$$

$$SS_{tot} = \sum_{i=1}^{n}(y_i - \bar{y})^2 \tag{2.16}$$

Dimana nilai R^2 adalah hasil perhitungan R^2, SS_{res} adalah total kuadrat selisih antara nilai aktual dan nilai prediksi, SS_{tot} adalah total kuadrat selisih antara nilai aktual dan rata-rata nilai aktual, y_i adalah nilai aktual, $\hat{y}_i$ adalah nilai prediksi, dan $\bar{y}$ adalah rata-rata nilai aktual. Metrik nilai R^2 ini intuitif dan mudah dipahami, sehingga dapat memberikan gambaran yang jelas mengenai seberapa baik model bekerja, namun metrik ini juga sensitif terhadap *outlier* sehingga bisa memberikan gambaran yang salah tentang kinerja model.

2.6.4 *Mean Absolute Percentage Error* (MAPE)

Mean Absolute Percentange Error adalah salah satu metrik evaluasi yang sering digunakan dalam model regresi, terutama ketika ingin mengetahui seberapa besar kesalahan prediksi dalam bentuk persentase. MAPE mengukur rata-rata kesalahan dari selisih absolut antara nilai prediksi dan nilai aktual yang dinyatakan sebagai persentase dari nilai aktual. Rumus perhitungan MAPE ditunjukkan pada Rumus 2.17.

$$\mathrm{Q} = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \tag{2.17}$$

Dimana Q adalah hasil perhitungan MAPE, y_i adalah nilai aktual, $\hat{y}_i$ adalah nilai yang diprediksi oleh model, dan n adalah jumlah *subset* data. Karena MAPE mengukur kesalahan dalam bentuk persentase, metrik ini memudahkan interpretasi performa model tanpa bergantung pada satuan data. Semakin kecil nilai MAPE, maka kualitas prediksi model semakin tinggi. Namun, MAPE bisa menjadi tidak stabil apabila terdapat nilai aktual yang sangat kecil (mendekati nol), karena akan menghasilkan pembagian dengan angka yang sangat kecil, sehingga kesalahan

tampak sangat besar. MAPE cocok digunakan pada *dataset* yang tidak mengandung nilai nol atau mendekati nol dalam nilai aktualnya (Tofallis, 2021).

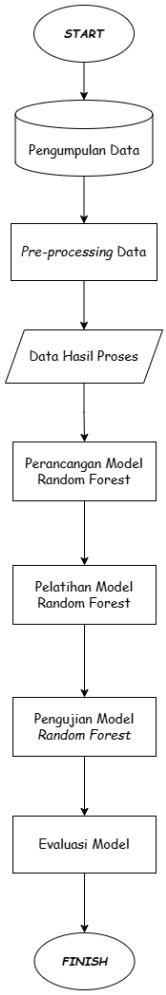
BAB III

METODE PENELITIAN

Dalam penelitian ini, digunakan pendekatan kuantitatif yang berfokus pada analisis data numerik yang diperoleh dari *dataset* film. Dengan menggunakan metode ini, hipotesis akan diuji melalui pengolah dan analisis data statistik menggunakan algoritma *machine learning*, yaitu *random forest*.

3.1 Desain Sistem

Desain sistem pada penelitian ini diawali dengan pengumpulan data film yang telah dirilis dari tahun 2010 hingga 2024. Setelah data terkumpul, dilakukan *pre-processing* yang mencakup *cleaning* data untuk memastikan kualitas data, termasuk menghapus fitur yang tidak relevan, konversi nilai untuk mengubah data ke dalam format numerik, kemudian dilakukan *handle outlier* untuk mengurangi dampak nilai ekstrem, serta standarisasi guna memastikan skala data konsisten. Setelah proses ini selesai, data dibagi menjadi dua bagian untuk *training* dan *testing* dengan tiga skenario, yaitu dengan rasio 9:1, 8:2, dan 7:3. Data *training* kemudian dilatih, dan algoritma *random forest* diimplementasikan dengan data yang ada. Dengan ini, model akan menghasilkan prediksi pendapatan film. Kemudian, model dievaluasi untuk menilai performa algoritma dalam memprediksi pendapatan film. Proses penelitian ini ditampilkan pada Gambar 3.1.



Gambar 3.1 Desain Sistem Penelitian

3.2 Pengumpulan Data

Data dalam penelitian ini merupakan data primer yang dikumpulkan secara manual dari tiga sumber utama, yaitu situs IMDb, *Box Office* Mojo, dan *YouTube*. Data yang diperoleh mencakup 500 film yang dirilis dari tahun 2010 hingga 2024. *Dataset* ini memiliki 7 fitur terkait informasi film, seperti judul, anggaran produksi, jumlah tayangan *trailer*, dan lainnya. Rincian fitur tersebut tercantum pada Tabel

3.1, dan Tabel 3.2 memperlihatkan contoh sampel *dataset* yang digunakan dalam penelitian ini.

Tabel 3.1 Detail Fitur *Dataset*

Fitur	Keterangan
Title	Judul dari suatu film
Overview	Ringkasan cerita suatu film
Release Date	Tanggal rilis film
Budget	Anggaran film dalam USD
Actor	Kategori ketenaran pemain film, terbagi menjadi 2 (terkenal dan kurang terkenal)
Views of Trailer	Jumlah penayangan *trailer* film
Revenue	Pendapatan asli film yang digunakan sebagai target perhitungan (USD)

Tabel 3.2 Contoh *Dataset* Film

No.	***Title***	***Overview***	***Release Date***	***Budget* (USD)**	***Actor***	***Trailer Views***	***Revenue* (USD)**
1	*Parasite*	*All unemployed, Ki-taek's family takes peculiar interest in the wealthy and glamorous Parks for their livelihood until they get entangled in an unexpected incident.*	30/05/2019	10.000.200	Terkenal	19.000.000	262.130.000
2	*Your Name.*	*High schoolers Mitsuha and Taki are complete strangers living separate lives. But one night, they suddenly switch places. Mitsuha wakes up in Takiâ€™s body, and he in hers. This bizarre occurrence continues to*	26/08/2016	5.000.800	Terkenal	9.400.000	382.240.000

Tabel 3.2 Contoh *Dataset* Film

No.	***Title***	***Overview***	***Release Date***	***Budget* (USD)**	***Actor***	***Trailer Views***	***Revenue* (USD)**
		happen randomly, and the two must adjust their lives around each other.					
3	*Spider-Man: Across the Spider-Verse*	*After reuniting with Gwen Stacy, Brooklynâ€™s full-time, friendly neighborhood Spider-Man is catapulted across the Multiverse, where he encounters the Spider Society, a team of Spider-People charged with protecting the Multiverseâ€™s very existence. But when the heroes clash on how to handle a new threat, Miles finds himself pitted against the other Spiders and must set out on his own to save those he loves most.*	31/05/2023	150.000.000	Terkenal	37.000.000	690.825.000

3.3 *Pre-processing*

Pre-processing adalah langkah awal yang berperan penting dalam proses analisis data. Pada tahap ini, dilakukan beberapa langkah guna menjamin bahwa data yang digunakan telah siap untuk dianalisis lebih lanjut. Proses ini mencakup

cleaning data, seperti menghapus fitur yang tidak relevan dan menangani data yang hilang, konversi nilai untuk mengubah data ke format yang dapat digunakan oleh model, serta *handle outlier* guna mengurangi dampak nilai ekstrem. Selanjutnya, dilakukan standarisasi untuk memastikan skala data konsisten. Tujuan dilakukan tahapan ini yaitu untuk meningkatkan kualitas data sehingga menghasilkan model yang akurat dan dapat diandalkan.

3.3.1 *Cleaning* Data

Langkah pertama dalam *pre-processing* adalah *cleaning* data, atau pembersihan data, dengan tujuan untuk memastikan kualitas dan konsistensi sebelum dianalisis lebih lanjut. Proses ini mencakup pengecekan dan penanganan nilai-nilai yang mungkin hilang (*missing values*) di setiap fitur, serta penghapusan fitur yang tidak relevan. Dalam penelitian ini, dilakukan pengecekan terhadap setiap fitur untuk melihat apakah terdapat data yang hilang atau tidak valid. Hasil pengecekan menunjukkan bahwa tidak ditemukan *missing values* maupun data yang tidak valid. Selain itu, fitur-fitur yang dianggap tidak memiliki kontribusi signifikan terhadap analisis dihapus agar model lebih efisien dan akurat.

3.3.2 Transformasi Data

Transformasi data dilakukan untuk mengonversi tipe data pada fitur tertentu agar dapat dianalisis dengan lebih optimal. Proses ini dilakukan setelah tahap *cleaning* data, sehingga fitur yang digunakan dalam penelitian ini telah melewati tahapan penyaringan.

Dalam *dataset* ini, semua fitur awalnya memiliki tipe data *object*, yang membuat analisis selanjutnya sulit dilakukan. Oleh karena itu, beberapa fitur perlu dikonversi menjadi tipe data numerik untuk memudahkan analisis. Proses transformasi ini mencakup perubahan tipe data pada fitur `budget_usd`, `trailer_views`, dan `revenue` menjadi numerik. Selain itu, fitur `actor` yang terdiri dari dua kategori, dikonversi menggunakan *label encoding*, dimana kategori "terkenal" direpresentasikan dengan nilai 1, dan kategori "kurang terkenal" dengan nilai 0. Setelah melakukan transformasi ini, *dataset* menjadi lebih terstruktur dan siap digunakan dalam pemodelan prediktif. Tipe data sebelum dan sesudah konversi ditampilkan dalam Tabel 3.3.

Tabel 3.3 Penyesuaian Tipe Data Fitur *Dataset*

Fitur	Tipe Data Lama	Tipe Data Baru
`budget_usd`	*object*	*int64*
`actor`	*object*	*int32*
`trailer_views`	*object*	*int64*
`revenue`	*object*	*int64*

3.3.3 *Handle Outlier*

Dalam penelitian ini, dilakukan deteksi dan penanganan *outlier* sebelum data digunakan untuk pelatihan model. *Outlier* diidentifikasi menggunakan IQR, dimana data yang berada di luar batas Q1 - 1.5 × IQR hingga Q3 + 1.5 × IQR dikategorikan sebagai *outlier* dan perlu ditangani agar tidak memengaruhi performa model. Setelah dilakukan identifikasi, *outlier* ditangani menggunakan teknik *clipping*, yaitu dengan membatasi nilai data dalam rentang IQR agar tidak ada nilai ekstrem yang dapat mengganggu proses pemodelan. Tabel 3.4 menampilkan jumlah *outlier* sebelum dan setelah dilakukan *handling*.

Tabel 3.4 Jumlah *Outlier* Sebelum dan Sesudah *Handling*

Fitur	Jumlah *Outlier*	
	Sebelum	Sesudah
budget_usd	9	0
trailer_views	50	0
revenue	32	0

Setelah menerapkan metode IQR, seluruh *outlier* berhasil dikurangi hingga nol. Proses *clipping* membatasi nilai-nilai ekstrem dalam rentang wajar, sehingga distribusi data menjadi lebih stabil dan siap untuk digunakan dalam analisis lebih lanjut.

3.3.4 Standarisasi Data

Setelah data melalui tahap penanganan *outlier*, langkah selanjutnya adalah standarisasi data. Proses ini menggunakan `StandardScaler`, yaitu metode yang mengubah data ke dalam skala dengan rata-rata (*mean*) 0 dan standar deviasi 1. Langkah ini diperlukan agar seluruh fitur memiliki skala yang seragam, sehingga model prediksi dapat berfungsi secara optimal. Salah satu efek dari standarisasi ini adalah hasil data yang dapat bernilai negatif. Hal ini terjadi karena `StandardScaler` melakukan transformasi berdasarkan nilai rata-rata dari data, yang berarti data dengan nilai di bawah rata-rata akan mendapatkan nilai negatif. Meskipun demikian, perubahan skala ini tidak mengubah hubungan antar data dan justru membantu model bekerja lebih baik dalam proses *training*.

3.4 Pembagian Data (*Data Splitting*)

Setelah melalui tahap *pre-processing*, data yang telah siap digunakan dalam analisis numerik dibagi menjadi dua bagian, yaitu data *training* dan data *testing*.

Data *training* berfungsi untuk membangun model, sementara data *testing* dimanfaatkan untuk menilai kemampuan model dalam menghadapi data yang belum dilihat sebelumnya. Guna mengevaluasi performa model secara lebih menyeluruh dalam berbagai kondisi, pembagian data dilakukan dalam tiga skenario dengan rasio yang berbeda:

1. 90 : 10 → 90 % data untuk pelatihan, 10% data untuk pengujian.
2. 80 : 20 → 80% data untuk pelatihan, 20% data untuk pengujian.
3. 70 : 30 → 70% data untuk pelatihan, 30% data untuk pengujian.

Setiap skenario ini digunakan untuk mengevaluasi bagaimana perubahan proporsi data latih dan data uji dapat memengaruhi model. Dengan mencoba berbagai skenario, dapat diperoleh gambaran mengenai stabilitas dan generalisasi model, serta menemukan rasio pembagian data yang paling optimal untuk analisis ini.

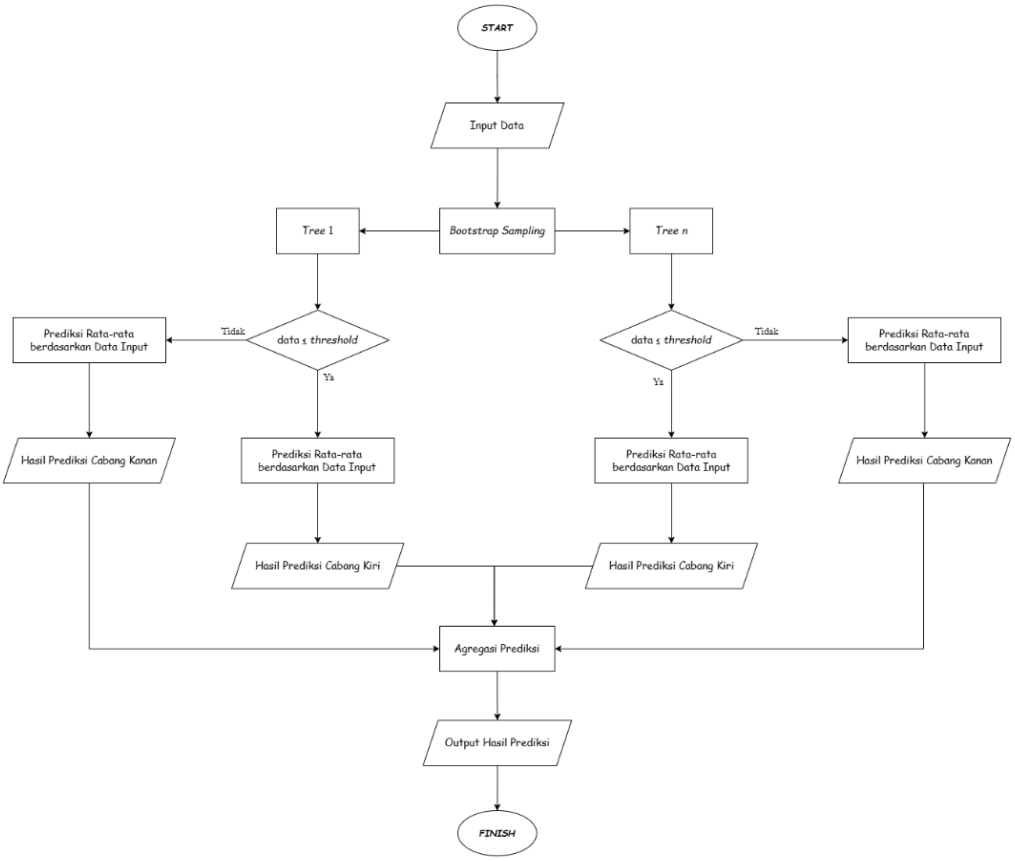
3.5 Implementasi *Random Forest*

Setelah proses pengolahan data selesai, langkah selanjutnya adalah mengimplementasikan metode *random forest* untuk memprediksi pendapatan film berdasarkan fitur anggaran, pemain, dan jumlah penayangan *trailer*. *Random Forest* merupakan metode *ensemble* dalam *machine learning* yang menyatukan beberapa *decision tree* guna meningkatkan akurasi dan stabilitas prediksi. Setiap pohon dalam *random forest*, akan menghasilkan hasil prediksi sendiri. Hasil akhir dari prediksi ditentukan berdasarkan rata-rata prediksi dari semua pohon, membuat algoritma ini kuat terhadap *overfitting* dan lebih stabil dibandingkan hanya menggunakan satu *decision tree*.

Proses implementasi *random forest* terdiri dari beberapa langkah, diantaranya adalah:

1. *Bootstrap sampling* – proses ini mengambil sampel data latih secara acak untuk setiap pohon yang akan dibangun. Teknik ini dikenal sebagai *bagging* atau *bootstrap aggregating* dan digunakan untuk meningkatkan variasi data di setiap pohon, sehingga setiap pohon memiliki data yang sedikit berbeda sebagai basisnya.
2. Pembentukan pohon keputusan – setelah terbentuk sampel data, setiap pohon keputusan dibangun dengan membuat aturan pemisahan (*splitting rules*) berdasarkan *subset* fitur yang dipilih secara acak. Pemilihan fitur secara acak ini membantu memastikan pohon-pohon tersebut tidak menjadi identik. Pohon akan berhenti membangun cabang baru ketika mencapai kedalaman maksimal (`max_depth`) atau ketika data dalam cabang tersebut homogen.
3. Agregasi hasil prediksi – setiap pohon dalam *random forest* menghasilkan prediksi terhadap nilai target, yaitu pendapatan film (`revenue`). Prediksi akhir diperoleh melalui agregasi, dengan cara menghitung rata-rata seluruh prediksi yang dihasilkan oleh setiap pohon dalam *random forest*.

Untuk memberikan gambaran yang lebih jelas mengenai alur kerja model *random forest*, Gambar 3.2 menyajikan *flowchart* yang menggambarkan alur implementasi model tersebut.



Gambar 3.2 *Flowchart* Implementasi *Random Forest*

Beberapa parameter penting juga digunakan dalam implementasi ini, yaitu:

- `n_estimators`, yaitu jumlah pohon yang dibangun dalam model, dimana jumlah pohon ini disesuaikan untuk mendapatkan keseimbangan antara akurasi dan efisiensi komputasi.
- `max_depth`, yaitu kedalaman maksimal dari setiap pohon, yang berguna untuk mencegah *overfitting* dengan menjaga agar pohon tidak terlalu kompleks.
- `min_samples_split`, yaitu jumlah sampel terkecil yang diperlukan dalam suatu *node* agar pemisahan dapat dilakukan, yang digunakan untuk mengontrol kompleksitas model.

- `min_samples_leaf`, yaitu jumlah sampel terkecil dalam setiap *leaf node* untuk mencegah model terlalu sensitif terhadap data *training*.
- *bootstrap*, yaitu metode pengambilan sampel dengan pengembalian yang digunakan untuk meningkatkan generalisasi model.
- `random_state`, yaitu nilai acak untuk memastikan hasil eksperimen dapat direproduksi, sehingga hasilnya akan konsisten setiap kali model dijalankan.

Sebagai contoh, misalkan data yang dimiliki yaitu data berisi fitur anggaran (*Budget*), apakah aktor terkenal atau kurang terkenal (*Actor*), dan jumlah penayangan *trailer* (*Trailer Views*) untuk memprediksi pendapatan film (*Revenue*). Tabel 3.5 menunjukkan permisalan data yang digunakan dalam penelitian.

Tabel 3.5 Contoh Data Permisalan

Film	***Budget* (juta)**	***Actor***	***Trailer Views* (juta)**	***Revenue* (target)**
A	100	1	5	120
B	150	0	10	80
C	200	1	20	250
D	120	0	15	100
E	180	1	12	210

1) Melakukan *bootstrap sampling* dengan dua contoh *bootstrap* untuk dua pohon awal.
   - *Tree* 1 : Sampel acak yang diambil yaitu film A, film C, dan film D
   - *Tree* 2 : Sampel acak yang diambil yaitu film B, film C, dan film E
2) Pembentukan *decision tree*, misalnya pada *tree* 1 terbentuk dengan menggunakan fitur "*Budget*" sebagai aturan pemisahan pertama. Data dibagi dengan berdasarkan apakah anggaran film lebih dari 150 juta atau tidak.
   - Cabang kiri, jika *budget* film adalah ≤ 150 juta, maka data yang masuk di cabang ini terdiri dari film A dan D. Berdasarkan data, *revenue* dari

film A adalah 120 juta, dan film D adalah 100 juta, maka hasil prediksi *revenue* cabang kiri pada *tree* 1 sebesar 110 juta.

$$\text{Prediksi cabang kiri} = \frac{120 + 100}{2} = 110 \text{ juta}$$

- Cabang kanan, jika *budget* film > 150 juta, maka data yang masuk di cabang ini adalah film C, dengan *revenue* sebesar 250 juta. Karena hanya ada satu film disini, maka *tree* 1 akan memprediksi *revenue* sebesar 250 juta untuk film dengan anggaran di atas 150 juta.

$$\text{Prediksi cabang kanan} = \frac{250}{1} = 250 \text{ juta}$$

Misalnya, pada *tree* 2 menggunakan fitur "*Trailer Views*" (jumlah penayangan *trailer*) untuk pemisahan. Data dibagi berdasarkan apakah *trailer views* film lebih dari 10 juta atau tidak.

- Cabang kiri, jika *trailer views* film ≤ 10 juta, maka data yang masuk di cabang ini hanya mencakup film B dengan *revenue* sebesar 80 juta. Karena hanya ada satu film disini, maka *tree* 2 akan memprediksi *revenue* sebesar 80 juta untuk film dengan *trailer views* dari 10 juta ke bawah.

$$\text{Prediksi cabang kiri} = \frac{80}{1} = 80 \text{ juta}$$

- Cabang kanan, jika *trailer views* film > 10 juta, maka data yang masuk di cabang ini terdiri dari film C, dengan *revenue* 250 juta, dan film E dengan *revenue* 210 juta. *Tree* 2 akan memprediksi *revenue* sebesar 230 juta untuk film dengan *trailer views* di atas 10 juta.

$$\text{Prediksi cabang kanan} = \frac{250 + 210}{2} = 230 \text{ juta}$$

3) Agregasi hasil prediksi dengan data baru, misalnya film baru memiliki *budget* 170 juta, *actor* terkenal (1), dan *trailer views* 14 juta, maka dapat dilakukan perhitungan prediksi.

- *Tree* 1, masuk di cabang kanan dengan *budget* 170 > 150, maka prediksi *revenue* 250 juta.
- *Tree* 2, masuk di cabang kanan dengan *trailer views* 15 > 10, maka prediksi *revenue* 230 juta.
- Prediksi akhir = $\frac{250 + 230}{2}$ = 240 juta

3.6 Evaluasi Model

Evaluasi model dilakukan untuk mengukur seberapa baik model yang telah dibangun. Dalam penelitian ini, evaluasi model dilakukan menggunakan tiga metrik, yaitu *Mean Absolute Error* (MAE), *Mean Squared Error* (MSE), dan koefisien determinasi (R^2). Sebagai upaya untuk mempermudah pemahaman, Gambar 3.3 menyajikan *flowchart* yang menggambarkan alur perhitungan ketiga metrik tersebut.

Gambar 3.3 *Flowchart* Evaluasi Model

3.6.1 *Mean Absolute Error* (MAE)

MAE berfungsi untuk menghitung rata-rata dari selisih absolut antara nilai prediksi dan nilai aktual, perhitungan ini menggunakan data contoh yang telah dihitung dan disajikan pada subbab sebelumnya. Pehitungan MAE dilakukan sesuai dengan rumus 2.12 yang telah dijelaskan pada bab 2.

$$Q = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

$$= \frac{1}{5}\left(|120 - 110| + |80 - 80| + |250 - 240| + |100 - 110| + |210 - 230|\right)$$

$$= \frac{1}{5}\left(10 + 0 + 10 + 10 + 20\right)$$

$$= \frac{50}{5}$$

$$= 10$$

3.6.2 *Mean Squared Error* (MSE)

MSE berfungsi untuk menghitung rata-rata dari kuadrat selisih antara nilai prediksi dan nilai aktual, perhitugan ini dilakukan sesuai dengan rumus 2.13 yang telah dijelaskan pada bab 2.

$$Q = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$= \frac{1}{5}\left((120-110)^2 + (80-80)^2 + (250-240)^2 + (100-110)^2 + (210-230)^2\right)$$

$$= \frac{1}{5}(100 + 0 + 100 + 100 + 400)$$

$$= \frac{700}{5}$$

$$= 140$$

3.6.3 Koefisien Determinasi (R^2)

Koefisien determinasi atau R^2 berfungsi untuk mengukur sejauh mana model dapat menjelaskan variasi yang terdapat dalam data. Perhitungan evaluasi ini mengacu pada rumus 2.14, 2.15, dan 2.16 yang telah dijelaskan dalam bab 2. Perhitungan pertama yang dilakukan yaitu menghitung rata-rata $\bar{y}$.

$$\bar{y} = \frac{y_1 + y_2 + y_3 + \ldots + y_n}{n}$$

$$= \frac{120 + 80 + 250 + 100 + 210}{5}$$

$$= \frac{760}{5}$$

$$= 152$$

Kemudian, setelah menemukan $\bar{y}$ = 152, dilakukan perhitungan SS_{res} dan SS_{tot}.

$$SS_{res} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$= (120 - 110)^2 + (80 - 80)^2 + (250 - 240)^2 + (100 - 110)^2 + (210 - 230)^2$

$= 100 + 0 + 100 + 100 + 400$

$= 700$

$SS_{tot} = \sum_{i=1}^{n}(y_i - \bar{y})^2$

$= (120 - 152)^2 + (80 - 152)^2 + (250 - 152)^2 + (100 - 152)^2 + (210 - 152)^2$

$= 1024 + 5184 + 9604 + 2704 + 3364$

$= 21880$

Setelah menemukan hasil SS_{res} dan SS_{tot}, kemudian melakukan perhitungan R^2.

$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$

$R^2 = 1 - \frac{700}{21880}$

$= 1 - 0.032$

$= 0.968$

Dari sini didapatkan nilai R^2 adalah 0.968, yang mengindikasikan bahwa model ini mampu menjelaskan sekitar 96.8% variasi dalam data.

3.6.4 *Mean Absolute Percentage Error* (MAPE)

MAPE berfungsi untuk menghitung rata-rata selisih absolut antara nilai prediksi dan nilai aktual, yang dinyatakan dalam persentase. Perhitungan ini dilakukan sesuai dengan rumus 2.17 yang telah dijelaskan pada bab 2.

$Q = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|$

$$= \frac{100\%}{5}\left(\left|\frac{120-110}{120}\right| + \left|\frac{80-80}{80}\right| + \left|\frac{250-240}{250}\right| + \left|\frac{100-100}{100}\right| + \left|\frac{210-230}{210}\right|\right)$$

$$= \frac{100\%}{5}(0.0833 + 0 + 0.04 + 0.10 + 0.0952)$$

$$= \frac{100\%}{5}(0.3185)$$

$$= 6.37\%$$

3.7 Skenario Pengujian

Tahap pertama dalam pengujian ini yaitu data dibagi menjadi dua bagian utama, yaitu fitur sebagai *input* (x) dan target (y). Pembagian ini dilakukan untuk mempersiapkan data agar dapat dianalisis lebih lanjut. Kolom target yang berisi pendapatan yaitu `revenue` akan diproses sebagai variabel target yang akan diprediksi oleh model. Kolom-kolom fitur lainnya akan mengalami transformasi agar dapat diproses oleh model prediksi. Sebagai contoh, kolom `actor` yang memiliki dua kategori yaitu terkenal dan kurang terkenal akan dilakukan pelabelan menggunakan *label encoding*, dimana label 1 menunjukkan kategori terkenal, dan label 0 menunjukkan kategori kurang terkenal. Proses ini dilakukan agar variabel kategorikal dapat dipahami dalam format numerik oleh model. Pada setiap skenario pengujian, parameter model yang digunakan adalah `n_estimators=500`, yaitu menggunakan 500 pohon keputusan dalam *ensemble* (*forest*), `max_depth=5`, yang membatasi kedalaman maksimal pohon hingga lima tingkat, agar model tidak menjadi terlalu rumit dan terjebak dalam *overfitting*, `min_samples_split=2`, `min_samples_leaf=4`, `bootstrap=True`, dan juga `random_state=42`. Selanjutnya, untuk melakukan uji coba, data dibagi menggunakan *train-test split*.

Persentase dari data pelatihan dan data pengujian pada setiap skenario disajikan dalam Tabel 3.6.

Tabel 3.6 Proporsi Data *Train* dan Data *Test*

Skenario	**Persentase Data *Training* (%)**	**Persentase Data *Testing* (%)**
A	90	10
B	80	20
C	70	30

Model A memanfaatkan 500 data yang dibagi menjadi 450 data *training* dan 50 data *testing*. Model B memanfaatkan jumlah data yang sama, yaitu 500, dengan komposisi 400 data *training* dan 100 data *testing*. Sedangkan model C memakai konfigurasi 350 data *training* dan 150 data *testing* dari total 500 data yang tersedia. Pembagian data ini membantu dalam pelatihan model menggunakan data *training* dan menguji performa model dengan data *testing*. Performa model diukur dengan menerapkan metrik MAE, MSE, dan R^2.

Pengujian ini juga dilakukan menggunakan teknik *K-Fold Cross Validation* untuk mengevaluasi performa model secara menyeluruh, dengan membagi data menjadi beberapa bagian (*fold*). Dalam penelitian ini, digunakan dua skenario validasi, yaitu 5-*Fold* dan 10-*Fold Cross Validation*. Tabel 3.7 menyajikan ilustrasi dari proses 5-*Fold Cross Validation*, sedangkan Tabel 3.8 menampilkan ilustrasi proses 10-*Fold Cross Validation*.

Tabel 3.7 Ilustrasi Proses 5-*Fold Cross Validation*

Fold	**5-*Fold Cross Validation***				
1	1 (*test*)	1 (*train*)	1 (*train*)	1 (*train*)	1 (*train*)
2	2 (*train*)	2 (*test*)	2 (*train*)	2 (*train*)	2 (*train*)
3	3 (*train*)	3 (*train*)	3 (*test*)	3 (*train*)	3 (*train*)
4	4 (*train*)	4 (*train*)	4 (*train*)	4 (*test*)	4 (*train*)
5	5 (*train*)	5 (*train*)	5 (*train*)	5 (*train*)	5 (*test*)

Tabel 3.8 Ilustrasi Proses 10-*Fold Cross Validation*

Fold	**10-*Fold Cross Validation***									
1	(*test*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)
2	(*train*)	(*test*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)
3	(*train*)	(*train*)	(*test*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)
4	(*train*)	(*train*)	(*train*)	(*test*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)
5	(*train*)	(*train*)	(*train*)	(*train*)	(*test*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)
6	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*test*)	(*train*)	(*train*)	(*train*)	(*train*)
7	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*test*)	(*train*)	(*train*)	(*train*)
8	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*test*)	(*train*)	(*train*)
9	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*test*)	(*train*)
10	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*train*)	(*test*)

Tabel 3.7 menampilkan proses 5-*Fold* yang diterapkan dengan nilai `cv=5`. Metrik evaluasi yang digunakan adalah R^2 untuk setiap *fold*, dan rata-rata nilai R^2 dari seluruh *fold* akan dihitung untuk memberikan gambaran performa model secara keseluruhan. Pada implementasi 5-*Fold*, parameter yang digunakan adalah `n_estimators=200`, `max_depth=5`, `min_samples_split=2`, untuk memastikan setiap *node* memiliki minimal 2 sampel sebelum dibagi, serta `min_samples_leaf=2` agar setiap daun memiliki minimal 2 sampel, dan `random_state=42` ditetapkan sehingga memberikan hasil yang konsisten pada setiap eksekusi. Kemudian, Tabel 3.8 menampilkan proses 10-*Fold* yang diterapkan dengan nilai `cv=10`. Parameter yang digunakan serupa dengan skenario sebelumnya, hanya saja jumlah pohon dikurangi, sehingga menggunakan `n_estimators=100`.

BAB IV

HASIL DAN PEMBAHASAN

Hasil-hasil pengujian model prediksi dipaparkan dalam bab ini. Proses dimulai dengan *pre-processing* data, diikuti dengan pembagian data menjadi *training* dan *testing*, serta penerapan tiga skenario pengujian manual dan validasi menggunakan K-*Fold Cross Validation*.

4.1 Uji Coba

Penelitian ini menggunakan *dataset* yang terdiri dari 500 data. Evaluasi performa model dilakukan melalui tiga skenario pengujian dimana data dibagi menjadi dua bagian, yaitu data data *training* dan data *testing*. Data *training* berperan dalam pembangunan model, sedangkan data *testing* berfungsi untuk mengukur sejauh mana model mampu melakukan prediksi dengan akurat. Pembagian persentase data untuk setiap skenario ditampilkan dalam Tabel 3.6 pada bab 3, sedangkan pembagian jumlah data *training* dan data *testing* tercantum di Tabel 4.1.

Tabel 4.1 Segmentasi *Dataset* Setiap Skenario

Skenario	**Data *Training***	**Data *Testing***
A	450	50
B	400	100
C	350	150

Setelah data dibagi, model dilatih menggunakan algoritma *random forest*, kemudian diuji untuk mengukur performanya berdasarkan metrik evaluasi MAE, MSE, dan R^2. Model divalidasi dengan dua skenario K-*Fold Cross Validation*,

yaitu 5-*Fold* dan 10-*Fold* untuk memastikan kestabilan dan keandalan hasil prediksi.

4.1.1 Implementasi Model pada Skenario Pengujian

Pada tahap ini, model prediksi menggunakan algoritma *Random Forest Regressor* dari *library* `scikit-learn`. Model dibangun untuk memprediksi nilai pendapatan (`revenue`) berdasarkan fitur *input* yang telah diproses sebelumnya. Data fitur dan data target dipersiapkan dengan kode yang terlihat pada Gambar 4.1.

```
x = np.hstack((scaled_features, data[['actor']].values))
y = scaled_revenue
```

Gambar 4.1 *Source Code* Pembagian Data *Input* dan Target

Gambar 4.1 menampilkan proses penggabungan fitur numerik yang telah distandarisasi, yaitu `budget_usd` dan `trailer_views`, dengan fitur kategorikal `actor` yang telah dikonversi ke bentuk numerik untuk membuat variabel *input* x. Sementara itu, variabel target y merepresentasikan nilai *revenue* yang telah distandarisasi. Setelah data dipersiapkan, model *Random Forest* dilatih menggunakan data *training* yang diperoleh dari pembagian data berdasarkan skenario pengujian. Model ini dibangun menggunakan beberapa parameter, antara lain jumlah pohon sebanyak 500, kedalaman maksimal tiap pohon sebesar 5, jumlah minimal sampel untuk pemisahan sebesar 2, dan jumlah minimal sampel pada daun sebesar 4. Setelah dilakukan proses *training*, model digunakan untuk melakukan prediksi terhadap data *testing*. Tahapan ini ditampilkan dalam Gambar 4.2.

```
model = RandomForestRegressor(n_estimators=500, max_depth=5,
                              min_samples_split=2, min_samples_leaf=4,
                              bootstrap=True, random_state=42)

model.fit(x_train, y_train)
prediction = model.predict(x_test)
```

Gambar 4.2 *Source Code* Model *Random Forest*

4.1.2 Validasi Model Menggunakan K-*Fold Cross Validation*

Metode K-*Fold Cross Validation* digunakan untuk memvalidasi model, dengan dua skenario, yaitu 5-*Fold* dan 10-*Fold*. Tujuannya untuk menguji stabilitas performa model pada pembagian data yang berbeda-beda. Sebelum melakukan validasi, model *Random Forest* untuk skenario 5-*Fold Cross Validation* dibangun dengan parameter seperti yang terlihat pada Gambar 4.3.

```
model_cv = RandomForestRegressor(n_estimators=200, max_depth=5,
                                 min_samples_split=2, min_samples_leaf=2,
                                 max_features='sqrt', bootstrap=True, random_state=42)
```

Gambar 4.3 *Source Code* Model *Random Forest* 5-*Fold Cross Validation*

Selanjutnya, proses validasi melibatkan pembagian data menjadi lima bagian (*fold*) yang secara bergantian digunakan sebagai data *testing*. Kode untuk proses 5-*Fold Cross Validation* bisa dilihat pada Gambar 4.4.

```
kf = KFold(n_splits=5, shuffle=True, random_state=42)
cv_scores = cross_val_score(model_cv, x, y, cv=kf, scoring='r2')
```

Gambar 4.4 *Source Code* 5-*Fold Cross Validation*

Kemudian, pada skenario 10-*Fold Cross Validation*, model *Random Forest* yang digunakan memiliki parameter yang sedikit berbeda, yaitu dengan `n_estimators=100`. Kode untuk proses pembangunan model ditunjukkan pada Gambar 4.5, dan kode proses validasinya dapat dilihat pada Gambar 4.6.

```
model_cv = RandomForestRegressor(n_estimators=100, max_depth=5,
                                 min_samples_split=2, min_samples_leaf=2,
                                 max_features='sqrt', bootstrap=True, random_state=42)
```

Gambar 4.5 *Source Code* Model *Random Forest* 10-*Fold Cross Validation*

```
kf = KFold(n_splits=10, shuffle=True, random_state=42)
cv_scores = cross_val_score(model_cv, x, y, cv=kf, scoring='r2')
```

Gambar 4.6 *Source Code* 10-*Fold Cross Validation*

4.2 Hasil Uji Coba

Hasil dari skenario uji coba yang dijabarkan pada subbab 4.1 dipaparkan dalam bagian ini. Pengujian dilakukan untuk mengevaluasi performa model dalam memprediksi hasil berdasarkan data yang dibagi sesuai skenario uji coba. Setiap skenario menerapkan fitur yang sama sebagai variabel *input* dalam pemodelan, yaitu anggaran produksi, ketenaran pemain, dan jumlah penayangan *trailer*.

Pengujian ini mencakup tiga skenario berbeda sesuai dengan pembagian data yang telah ditentukan. Model dievaluasi menggunakan MAE, MSE, dan R^2, serta MAPE sebagai tambahan untuk pengecekan. Selain itu, guna memastikan kestabilan dan keandalan model, dilakukan juga pengujian validasi dengan dua skenario K-*Fold Cross Validation*.

4.2.1 Hasil Uji Coba Berdasarkan Pembagian Data

Proses pengujian memanfaatkan tiga skenario pembagian data, yaitu 9:1, 8:2, dan 7:3. Setiap skenario bertujuan untuk melihat bagaimana model beradaptasi dengan jumlah data *training* yang berbeda dan mengevaluasi kinerjanya dalam melakukan prediksi. Hasil yang ditampilkan telah melalui proses standarisasi, sehingga nilainya berada dalam skala yang seragam.

Pada pengujian pertama, data dibagi dengan rasio 9:1, menghasilkan 450 data pelatihan dan 50 data pengujian. Hasil prediksi model pada pengujian ini tertera pada Tabel 4.2.

Tabel 4.2 Hasil Prediksi Skenario A

Skenario A (9:1)		
Data	**Hasil Prediksi**	**Hasil Prediksi (USD)**
1	0.85202815	587.325.584
2	-0.67305945	83.885.575
3	1.22229904	709.554.090
4	-0.81889486	35.774.485
5	-0.81043013	38.538.739

Tabel 4.2 menampilkan lima hasil prediksi model terhadap data uji pada skenario A. Nilai prediksi yang ditampilkan mencakup dua bentuk, yaitu hasil estimasi pendapatan yang telah melalui proses standarisasi (*scaling*) dan hasil estimasi yang telah dikembalikan ke satuan asli pendapatan, yaitu USD. Pada skenario ini, evaluasi model menghasilkan nilai MAE sebesar 0.36, MSE sebesar 0.32, dan R^2 sebesar 0.6934.

Pada pengujian kedua, pembagian data dilakukan dengan rasio 8:2, dimana 400 data untuk *training* dan 100 data untuk *testing*. Hasil prediksi model pada skenario ini dapat dilihat pada Tabel 4.3.

Tabel 4.3 Hasil Prediksi Skenario B

Skenario B (8:2)		
Data	**Hasil Prediksi**	**Hasil Prediksi (USD)**
1	0.85438281	588.102.870
2	-0.63421339	96.708.876
3	1.36976343	758.232.915
4	-0.8222379	34.640.926
5	-0.05112434	289.189.857

Tabel 4.3 menampilkan lima hasil prediksi model pada skenario B. Pada skenario kedua ini, evaluasi kinerja model diperoleh dengan nilai MAE sebesar 0.41, MSE sebesar 0.37, dan R^2 sebesar 0.6550.

Pada pengujian ketiga, data dibagi dengan rasio 7:3, menghasilkan 350 data latih dan 150 data uji. Hasil prediksi model dalam skenario ini ditampilkan dalam Tabel 4.4.

Tabel 4.4 Hasil Prediksi Skenario C

Skenario C (7:3)		
Data	**Hasil Prediksi**	**Hasil Prediksi (USD)**
1	0.88515817	598.261.987
2	-0.5771546	115.544.306
3	1.4343391	779.549.709
4	-0.77594282	49.923.192
5	-0.78010828	48.548.150

Tabel 4.4 menyajikan lima nilai prediksi model pada skenario C. Pada skenario ini, didapatkan evaluasi model dengan nilai MAE sebesar 0.40, MSE sebesar 0.34, dan R^2 sebesar 0.6709.

4.2.2 Hasil Uji Coba dengan K-*Fold Cross Validation*

Pengujian performa model dilakukan secara menyeluruh menggunakan teknik *Cross Validation*. Metode yang digunakan adalah K-*Fold Cross Validation* dengan dua skenario, yaitu 5-*Fold* dan 10-*Fold*. Pada skenario 5-*Fold Cross Validation*, data dibagi menjadi lima bagian. Setiap *fold* secara bergantian berperan sebagai data *testing*, sementara empat *fold* lainnya berperan sebagai data *training*. Hasil evaluasi performa model diukur menggunakan nilai R^2 (R^2 *score*) pada

setiap *fold*. Nilai R^2 dari masing-masing *fold* pada skenario 5-*Fold Cross Validation* ditampilkan pada Tabel 4.5.

Tabel 4.5 Hasil Evaluasi Model dengan 5-*Fold Cross Validation*

	***Fold* 1**	***Fold* 2**	***Fold* 3**	***Fold* 4**	***Fold* 5**
R^2	0.66275674	0.71042238	0.61908139	0.643172	0.65010888

Dari hasil R^2 tiap *fold* pada Tabel 4.5, didapatkan rata-rata nilai R^2 sebesar 0.6571 dengan standar deviasi 0.0302. Performa model juga diukur menggunakan MAE dengan memperoleh nilai sebesar 0.41, dan MSE sebesar 0.34. Pengujian selanjutnya yaitu dengan skenario 10-*Fold Cross Validation*, dimana data dibagi menjadi sepuluh bagian. Nilai R^2 untuk setiap *fold* ditampilkan pada Tabel 4.6.

Tabel 4.6 Hasil Evaluasi Model dengan 10-*Fold Cross Validation*

***Fold* 1**	***Fold* 2**	***Fold* 3**	***Fold* 4**	***Fold* 5**	***Fold* 6**	***Fold* 7**	***Fold* 8**	***Fold* 9**	***Fold* 10**
0.7246	0.6138	0.8216	0.6433	0.5727	0.6800	0.5608	0.6394	0.5414	0.7161

Rata-rata nilai R^2 yang diperoleh dari skenario 10-*Fold Cross Validation* adalah 0.6468, dengan standar deviasi 0.0875. Sementara itu, diperoleh nilai MAE sebesar 0.41, dan MSE sebesar 0.34.

4.3 Pembahasan

Pengujian dilakukan dengan 500 data film yang diperoleh dari IMDb, *Box Office* Mojo, dan *YouTube* sebagai sumber utama. Data ini kemudian melalui tahap *pre-processing* yang mencakup *cleaning* data, konversi nilai, *handle outlier*, *label encoding*, dan juga standarisasi data menggunakan `StandarScaler`. Setelah proses ini, data dibagi menjadi tiga skenario berbeda untuk menguji performa model.

Hasil uji coba pada subbab 4.2 menunjukkan bahwa setiap skenario menghasilkan performa evaluasi yang berbeda. Pada uji coba pertama dengan rasio data *training* dan *testing* sebesar 9:1, memperoleh nilai MAE sebesar 0.36, MSE 0.32, dan R^2 0.6934. Pengujian kedua dilakukan dengan rasio 8:2 yang memperoleh nilai MAE sebesar 0.41, MSE 0.37, dan R^2 0.6550. Sementara itu, pada skenario ketiga dengan perbandingan 7:3 diperoleh nilai MAE sebesar 0.40, MSE 0.34, dan R^2 0.6709. Nilai dari hasil evaluasi performa model untuk masing-masing skenario dapat dilihat pada Tabel 4.7.

Tabel 4.7 Hasil Evaluasi Performa Model

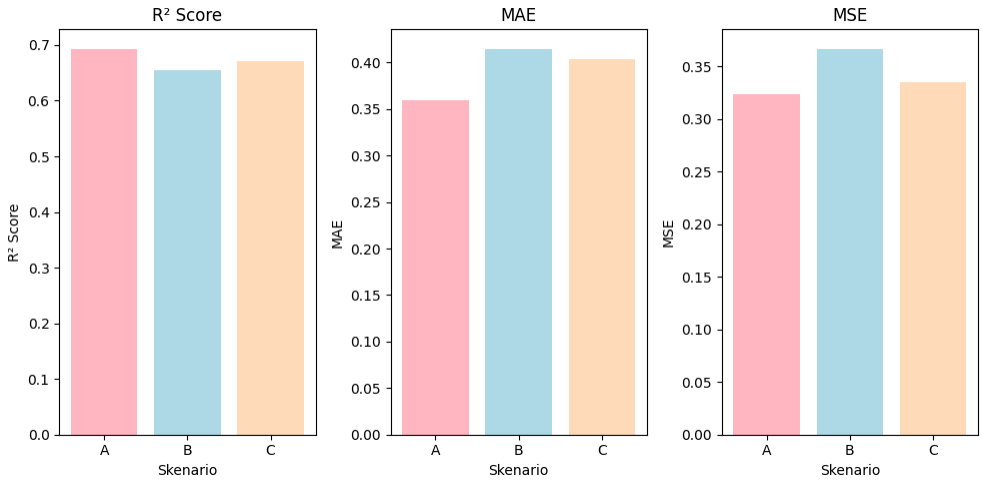
Skenario	**MAE**	**MSE**	$\boldsymbol{R^2}$
A	0.36	0.32	0.6934
B	0.41	0.37	0.6550
C	0.40	0.34	0.6709

Tabel 4.7 menampilkan perbandingan hasil evaluasi performa model pada tiga skenario yang telah diterapkan. MAE mencerminkan rata-rata kesalahan absolut antara nilai prediksi dan nilai aktual. Pada skenario A, yaitu rasio 9:1, MAE menghasilkan nilai terendah 0.36 yang menandakan bahwa model memiliki kesalahan rata-rata yang lebih kecil dibandingkan skenario lainnya. Sementara itu, pada skenario B yaitu rasio 8:2, nilai MAE meningkat menjadi 0.41, yang berarti model cenderung memiliki kesalahan prediksi yang lebih besar. Pada skenario C yaitu rasio 7:3, MAE sedikit lebih rendah dibanding skenario B, yaitu 0.40, tetapi masih lebih tinggi dibanding skenario A. Hal ini menunjukkan bahwa model cenderung bekerja lebih baik dengan data *training* yang lebih dominan, tetapi cukup stabil saat proporsi data *testing* meningkat.

MSE mengukur rata-rata kuadrat dari kesalahan prediksi, sehingga kesalahan yang lebih besar akan memiliki bobot lebih tinggi. Sama seperti MAE, semakin kecil nilai MSE, semakin baik performa model. Pada skenario A, MSE memiliki nilai sebesar 0.32, yang menunjukkan bahwa model dalam skenario ini lebih stabil dalam meminimalkan kesalahan prediksi. Skenario C memiliki nilai MSE sedikit lebih tinggi dari skenario A, yaitu sebesar 0.34, yang berarti model ini masih cukup baik, meskipun tingkat kesalahannya sedikit lebih tinggi dibandingkan skenario A. Sementara itu, skenario B memiliki nilai MSE tertinggi sebesar 0.37, yang menunjukkan bahwa model pada skenario ini mengalami lebih banyak kesalahan signifikan dibandingkan skenario lainnya. Hal ini menunjukkan bahwa distribusi data *training* yang lebih besar akan berkontribusi pada peningkatan kinerja model.

R^2 digunakan untuk mengukur sejauh mana model dapat menjelaskan variasi dalam data. Nilai R^2 yang lebih tinggi menunjukkan bahwa model lebih mampu menangkap pola hubungan antara variabel *input* dan *output*. Pada skenario A, R^2 memiliki nilai tertinggi yaitu 0.6934 yang menunjukkan bahwa model mampu menjelaskan sekitar 69.34% variabilitas data. Pada skenario C, nilai R^2 sedikit lebih rendah yaitu 0.6709, yang masih menunjukkan performa yang cukup baik. Sementara pada skenario B, didapatkan nilai R^2 terendah yaitu 0.6550, yang mengindikasikan bahwa performa prediksi model pada skenario ini lebih rendah dari pada skenario lainnya.

Perbandingan hasil evaluasi performa model dari ketiga skenario berdasarkan nilai MAE, MSE, dan R^2 dapat dilihat secara lebih jelas pada Gambar 4.7.



Gambar 4.7 Visualisasi Hasil Evaluasi Performa Model

Gambar 4.7 menunjukkan perbandingan nilai MAE, MSE, dan R^2 untuk ketiga skenario yang diuji. Skenario A memiliki performa terbaik dengan nilai MAE dan MSE paling rendah, serta nilai R^2 tertinggi, menunjukkan bahwa model dapat memprediksi dengan lebih baik dibanding skenario lainnya. Sementara itu, skenario B menunjukkan performa terendah dengan nilai MAE dan MSE tertinggi, serta R^2 terendah, menandakan model mengalami lebih banyak kesalahan prediksi. Skenario C berada di antara keduanya, dengan performa yang lebih stabil dibanding skenario B tetapi belum seoptimal skenario A. Dari hasil evaluasi performa model pada ketiga skenario ini, menunjukkan bahwa keseimbangan antara data *training* dan *testing* berpengaruh terhadap kualitas prediksi model.

Selain tiga metrik utama tersebut, model juga dievaluasi menggunakan MAPE sebagai metrik tambahan. Nilai MAPE pada skenario A tercatat sebesar 126.49%, skenario B sebesar 228.83%, dan skenario C sebesar 446.92%. Nilai MAPE yang tinggi mengindikasikan bahwa persentase kesalahan prediksi rata-rata terhadap nilai aktual cukup besar, terutama pada skenario C. Sebaliknya, skenario

A sebagai skenario dengan performa terbaik memiliki nilai MAPE paling rendah, yang menandakan prediksi model lebih mendekati nilai sebenarnya. Pada Tabel 4.8, ditampilkan dua data dari skenario A yang menunjukkan selisih prediksi terbesar dan terkecil terhadap nilai aktual.

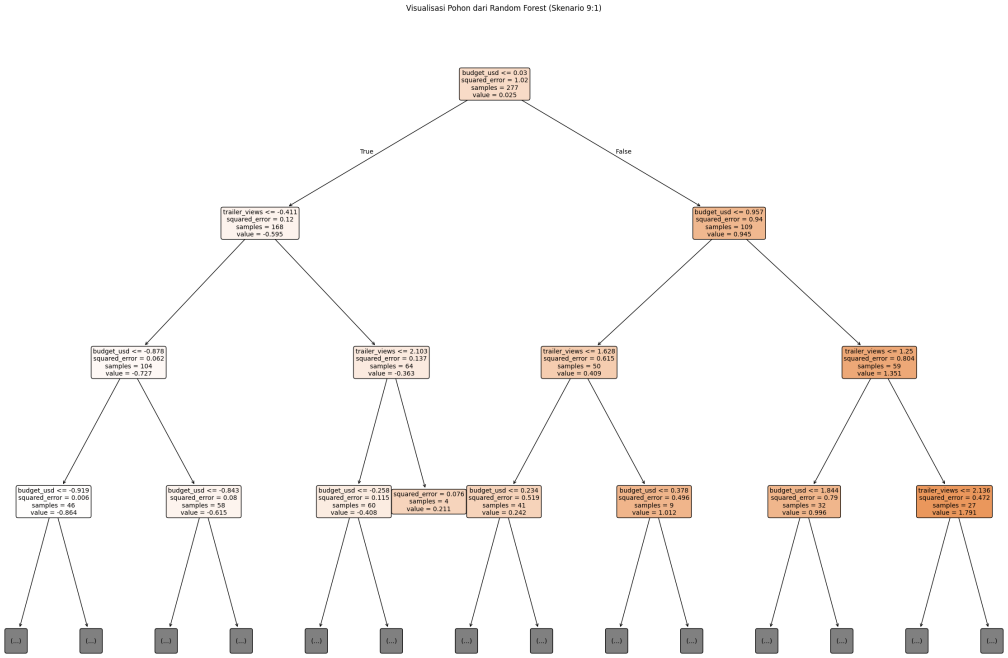
Tabel 4.8 Perbandingan Selisih Terbesar dan Terkecil pada Skenario A

Selisih	Data Ke-	Nilai Aktual (USD)	Prediksi (USD)	Selisih (USD)
Terbesar	45	1.078.959.000,00	366.241.383,47	712.717.616,53
Terkecil	48	92.560.000,00	92.667.918,83	107.918,83

Seperti terlihat pada Tabel 4.8, selisih prediksi terbesar terjadi pada data ke-45, dimana nilai prediksi berbeda cukup jauh dari nilai asli, dengan selisih sebesar $717.717.616,53. Sementara itu, selisih terkecil ditemukan pada data ke-48 dengan selisih hanya sebesar $107.918,93. Penyebab besarnya selisih pada data ke-45 adalah karena nilai aktualnya sangat tinggi dan termasuk data ekstrem yang ditangani menggunakan teknik *clipping*, sehingga prediksi model menjadi jauh lebih rendah dari nilai aslinya. Perbedaan yang cukup mencolok ini menunjukkan bahwa meskipun secara keseluruhan skenario A memberikan hasil prediksi yang paling mendekati nilai aktual, model masih memiliki potensi menghasilkan kesalahan prediksi yang cukup besar pada data tertentu. Adanya selisih prediksi yang cukup besar pada beberapa data menjelaskan mengapa nilai MAPE pada skenario A masih cukup tinggi, meskipun lebih baik dibandingkan dua skenario lainnya.

Sebagai bagian dari interpretasi model, dilakukan visualisasi salah satu pohon pada *Random Forest* dari skenario terbaik, yaitu rasio 9:1, untuk membantu

memahami struktur keputusan yang dibuat oleh model berdasarkan fitur-fitur yang digunakan. Visualisasi disajikan dalam Gambar 4.8.



Gambar 4.8 Visualisasi Pohon *Random Forest*

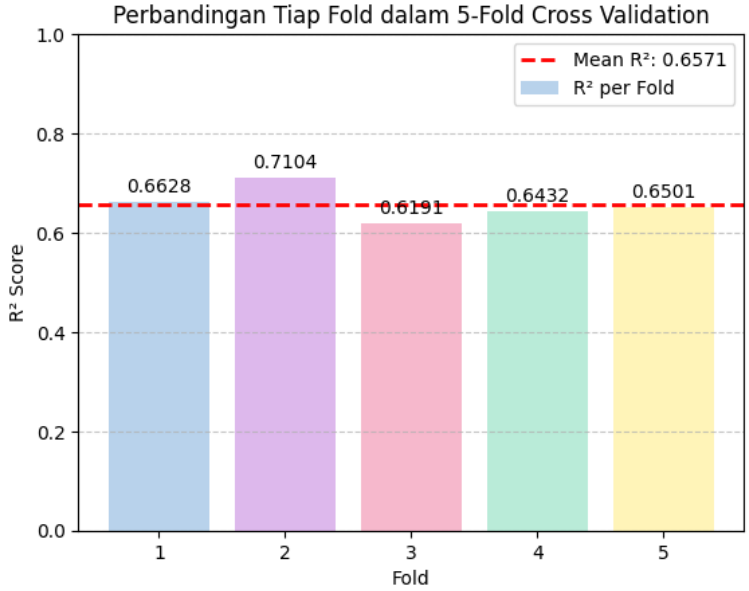
Gambar 4.8 menampilkan visualisasi salah satu pohon keputusan dari model *random forest* yang dilatih menggunakan skenario dengan rasio data 9:1, 90% data *training* dan 10% data *testing*. Pohon ini dibangun melalui proses pemilihan fitur terbaik pada setiap *node* untuk meminimalkan nilai kesalahan prediksi. Dalam pohon ini, terlihat bahwa dua fitur utama yang paling sering digunakan adalah `budget_usd` dan `trailer_views`. Hal ini menunjukkan bahwa kedua fitur tersebut memberikan kontribusi terbesar dalam menentukan hasil prediksi pada pohon yang ditampilkan. Struktur pohon dimulai dari *root node* (bagian paling atas), yang merupakan titik awal pembagian data. Pada *node* ini, data dibagi berdasarkan nilai `budget_usd` $\leq$ 0.03. Artinya, model memisahkan data

berdasarkan besar kecilnya nilai anggaran produksi dalam satuan standarisasi. Apabila kondisi terpenuhi, maka data masuk ke cabang kiri, sedangkan jika tidak, data masuk ke cabang kanan. Proses ini terus berlangsung di setiap level pohon berdasarkan fitur dan nilai *threshold* tertentu, hingga mencapai *leaf node*, yaitu titik akhir keputusan model.

Setiap *leaf node* menampilkan nilai `value`, yang merepresentasikan rata-rata nilai target dari seluruh data *training* yang mencapai *node* tersebut. Sementara itu, nilai `samples` menunjukkan jumlah data yang berada di dalam *node* tersebut. Semakin kecil nilai *varians*, maka semakin homogen data dalam *node* tersebut, yang menandakan bahwa pembagian sebelumnya telah mampu mengelompokkan data dengan karakteristik yang serupa. Pohon yang divisualisasikan ini dibatasi oleh pengaturan parameter `max_depth=3`, untuk membatasi kedalaman tampilan pohon. Tanda titik-titik pada bagian bawah menunjukkan bahwa struktur pohon masih berlanjut, namun tidak ditampilkan secara menyeluruh karena keterbatasan kedalaman tampilan.

Model juga dievaluasi menggunakan dua skenario K-*Fold Cross Validation*, yaitu 5-*Fold* dan 10-*Fold* untuk menguji kestabilan performa. Pada skenario 5-*Fold Cross Validation*, hasil evaluasi menunjukkan fluktuasi nilai R^2 pada setiap *fold* yang relatif kecil, seperti yang ditampilkan pada Tabel 4.5 dalam hasil uji coba. Secara keseluruhan, model menghasilkan rata-rata nilai R^2 sebesar 0.6571 dengan standar deviasi 0.0302. Standar deviasi ini mengindikasikan seberapa jauh nilai R^2 pada setiap *fold* menyimpang dari rata-rata, dimana semakin kecil nilainya, semakin konsisten performa model di setiap *fold*. Model juga menghasilkan nilai MAE

sebesar 0.41 dan MSE sebesar 0.34. Gambar 4.9 mengilustrasikan hasil evaluasi dengan menampilkan nilai R^2 untuk setiap *fold* dan rata-ratanya.



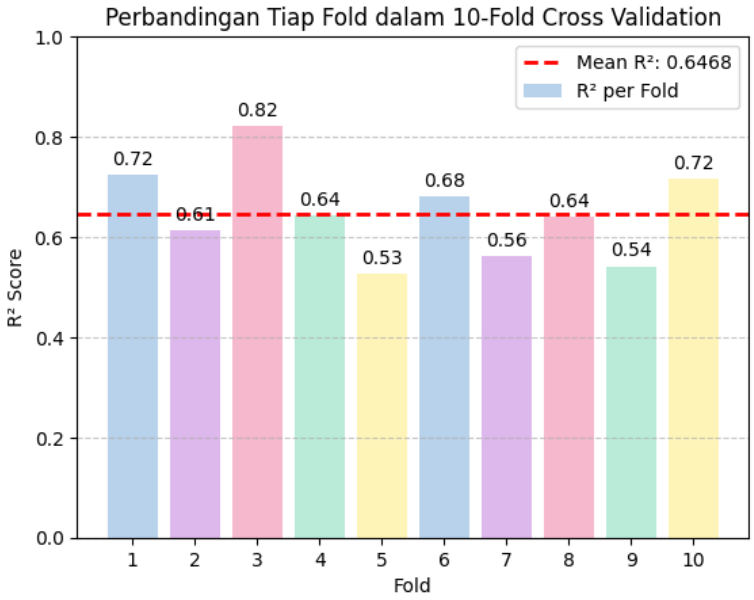
Gambar 4.9 Visualisasi Performa Model pada Tiap *Fold* dalam 5-*Fold*

Hasil evaluasi menunjukkan fluktuasi nilai R^2 pada setiap *fold*. *Fold* pertama memiliki nilai R^2 sebesar 0.6628, *fold* ke-dua mendapat nilai R^2 tertinggi sebesar 0.7104, menandakan kekuatan prediksi model yang optimal dalam *subset* ini, *fold* ke-tiga memperoleh nilai R^2 terendah yaitu 0.6191, menunjukkan bahwa model mengalami penurunan akurasi pada *subset* ini, *fold* ke-empat 0.6432, dan *fold* ke-lima memiliki nilai R^2 sebesar 0.6501. Dari hasil ini, dapat dilihat bahwa performa model paling tinggi terjadi pada *fold* ke-dua, sedangkan performa terendah terjadi pada *fold* ke-tiga. Fluktuasi ini menunjukkan bahwa kemampuan model dalam melakukan prediksi dapat bervariasi tergantung pada *subset* data yang digunakan dalam proses validasi.

Garis merah putus-putus pada grafik menunjukkan rata-rata nilai R^2 dari keseluruhan *fold*, yaitu 0.6571. *Fold* pertama dan ke-dua memiliki nilai di atas rata-

rata, sementara *fold* ke-tiga, ke-empat, dan ke-lima berada di bawahnya. Hal ini menandakan bahwa meskipun model memiliki performa yang cukup baik pada sebagian *fold*, terdapat ketidakseimbangan performa di beberapa *subset* data yang diuji. Hasil ini memberikan gambaran bahwa model masih memiliki ruang untuk perbaikan, terutama dalam meningkatkan kestabilan performa pada seluruh *fold* agar fluktuasi antar *subset* tidak terlalu signifikan.

Pada skenario 10-*Fold Cross Validation*, model menunjukkan fluktuasi nilai R^2 pada setiap *fold* yang cukup besar, sebagaimana ditampilkan pada Tabel 4.6 dalam hasil uji coba. Secara keseluruhan, rata-rata nilai R^2 yang diperoleh sebesar 0.6468 dengan standar deviasi 0.0875. Peningkatan nilai standar deviasi ini mengindikasikan adanya fluktuasi performa model antar *fold*, yang menunjukkan bahwa kestabilan model masih perlu ditingkatkan. Model juga menghasilkan nilai MAE sebesar 0.41 dan MSE sebesar 0.34. Gambar 4.10 menampilkan visualisasi nilai R^2 untuk setiap *fold* beserta rata-rata yang didapat dari skenario 10-*Fold*.



Gambar 4.10 Visualisasi Performa Model pada Tiap *Fold* dalam 10-*Fold*

Hasil evaluasi menunjukkan adanya fluktuasi nilai R^2 pada setiap *fold*. Pada *fold* pertama, model mencatat nilai R^2 sebesar 0.72, menunjukkan performa cukup baik. *Fold* ke-dua mengalami penurunan dengan nilai 0.62, sementara *fold* ke-tiga memperoleh nilai tertinggi sebesar 0.82, mencerminkan prediksi yang paling optimal di antara semua *subset*. *Fold* ke-empat menghasilkan nilai 0.64, dan *fold* ke-lima mencatat nilai terendah, yaitu 0.53, menunjukkan kesulitan model dalam melakukan generalisasi pada *subset* tersebut. *Fold* ke-enam memperoleh nilai 0.68, *fold* ke-tujuh 0.56, *fold* ke-delapan 0.64, *fold* ke-sembilan 0.54, dan *fold* ke-sepuluh kembali menunjukkan performa cukup tinggi dengan nilai 0.72. Fluktuasi nilai ini menunjukkan bahwa kemampuan model dalam melakukan prediksi dapat bervariasi tergantung pada *subset* data yang digunakan dalam proses validasi.

Rata-rata nilai R^2 dari keseluruhan *fold* ditunjukkan oleh garis merah putus-putus pada gambar, yaitu sebesar 0.6468. *Fold* pertama, ke-tiga, ke-enam, dan ke-sepuluh memiliki nilai R^2 di atas rata-rata, sementara enam *fold* lainnya berada di bawahnya. Hal ini menunjukkan bahwa meskipun model memiliki performa yang cukup baik pada beberapa *fold*, masih terdapat ketidakseimbangan performa hasil pengujian terhadapa *subset* data tertentu. Hasil ini memberikan gambaran bahwa model masih memiliki ruang untuk perbaikan, khususnya dalam upaya meningkatkan kestabilan performa pada seluruh *fold* agar variasi antar *subset* tidak terlalu signifikan.

Pada pengujian penelitian ini, model menunjukkan performa terbaik pada skenario A dengan rasio 9:1 dengan MAE paling kecil dan nilai R^2 paling tinggi, sementara skenario B dengan rasio 8:2 memiliki performa paling rendah. Hasil dari

dua skenario K-*Fold Cross Validation*, yaitu 5-*Fold* dan 10-*Fold* juga mengindikasikan adanya ketidakseimbangan performa pada beberapa *subset* data. Meskipun kedua skenario menunjukkan performa yang cukup baik, kestabilan model masih perlu ditingkatkan agar hasilnya lebih konsisten di berbagai *subset* data.

Dalam industri perfilman, prediksi pendapatan menjadi salah satu aspek penting dalam perencanaan strategi pemasaran dan distribusi. Dengan adanya model prediksi, industri film dapat memperkirakan performa suatu film sebelum dirilis, sehingga keputusan bisnis dapat diambil dengan lebih bijak. Konsep prediksi ini sejatinya telah ada sejak dahulu, sebagaimana yang dicontohkan dalam kisah Nabi Yusuf a.s. yang mampu menafsirkan mimpi raja mengenai tujuh tahun masa subur dan tujuh tahun masa sulit. Sebagaimana disebutkan dalam firman Allah, Surah Yusuf ayat 47-48, yang berbunyi:

قَالَ تَزْرَعُوْنَ سَبْعَ سِنِيْنَ دَاَبًاۚ فَمَا حَصَدْتُّمْ فَذَرُوْهُ فِيْ سُنْۢبُلِهٖٓ اِلَّا قَلِيْلًا مِّمَّا تَأْكُلُوْنَ ﴿٤٧﴾ ثُمَّ يَأْتِيْ مِنْۢ بَعْدِ ذٰلِكَ سَبْعٌ شِدَادٌ يَّأْكُلْنَ مَا قَدَّمْتُمْ لَهُنَّ اِلَّا قَلِيْلًا مِّمَّا تُحْصِنُوْنَ ﴿٤٨﴾

"*(Yusuf) berkata, "Bercocoktanamlah kamu tujuh tahun berturut-turut! Kemudian apa yang kamu tuai, biarkanlah di tangkainya, kecuali sedikit untuk kamu makan. Kemudian, sesudah itu akan datang tujuh (tahun) yang sangat sulit (paceklik) yang menghabiskan apa yang kamu simpan untuk menghadapinya, kecuali sedikit dari apa (bibit gandum) yang kamu simpan*" *(QS. Yusuf 12:47-48)*.

Dalam tafsir tahlili, kedua ayat ini menjelaskan bahwa Yusuf dengan sikap dermawan menerangkan arti mimpi raja, seakan-akan ia sedang menyampaikan kepada raja dan pembesar istana, dengan mengatakan, "Wahai raja dan pembesar-pembesar negara semuanya, kamu akan menghadapi suatu masa tujuh tahun lamanya penuh dengan segala kemakmuran dan keamanan. Ternak berkembang

biak, tumbuh-tumbuhan subur, dan semua orang akan merasa senang dan bahagia. Maka galakkanlah rakyat bertanam dalam masa tujuh tahun itu. Hasil dari tanaman itu harus kamu simpan, gandum disimpan dengan tangkai-tangkainya supaya tahan lama. Sebagian kecil kamu dikeluarkan untuk dimakan sekadar keperluan saja. Sehabis masa yang makmur itu, akan datang masa yang penuh kesengsaraan dan penderitaan selama tujuh tahun pula. Pada waktu itu ternak habis musnah, tanaman-tanaman tidak berbuah, udara panas, musim kemarau panjang. Sumber-sumber air menjadi kering dan rakyat menderita kekurangan makanan. Semua simpanan makanan akan habis, kecuali tinggal sedikit untuk kamu jadikan benih." (Musthofa, 2021).

Dalam konteks penelitian ini, metode prediksi yang digunakan memiliki konsep yang serupa, yaitu memperkirakan pendapatan suatu film sebelum perilisan agar strategi pemasaran dan distribusi dapat dirancang dengan lebih efektif. Seperti halnya Nabi Yusuf a.s. yang memberikan peringatan tentang tujuh tahun masa subur dan tujuh tahun masa sulit sehingga memungkinkan perencanaan yang lebih matang, model prediksi dalam penelitian ini berperan sebagai alat bantu bagi industri film dalam pengambilan keputusan berdasarkan data historis dan tren pasar. Dengan demikian, pihak industri dapat mengoptimalkan sumber daya secara lebih efisien untuk mencapai hasil yang diharapkan.

Hasil penelitian ini menunjukkan bahwa pendapatan film dapat diprediksi dengan mempertimbangkan data historis seperti anggaran produksi, ketenaran pemain, dan jumlah penayangan trailer. Dalam konteks ini, proses mempelajari masa lalu dan menghasilkan prediksi untuk masa depan bukan hanya merupakan

pendekatan ilmiah, namun juga sejalan dengan nilai-nilai Islam yang mendorong untuk memiliki perencanaan yang matang. Dalam Al-Qur'an Surah Al-Hasyr ayat 18, Allah berfirman:

يَٰٓأَيُّهَا ٱلَّذِينَ ءَامَنُواْ ٱتَّقُواْ ٱللَّهَ وَلۡتَنظُرۡ نَفۡسٞ مَّا قَدَّمَتۡ لِغَدٖۖ وَٱتَّقُواْ ٱللَّهَۚ إِنَّ ٱللَّهَ خَبِيرُۢ بِمَا تَعۡمَلُونَ

"*Hai orang-orang yang beriman, bertakwalah kepada Allah dan hendaklah setiap diri memperhatikan apa yang telah diperbuatnya untuk hari esok, dan bertakwalah kepada Allah, sesungguhnya Allah Maha Mengetahui apa yang kamu kerjakan*" *(QS. Al-Hasyr 59:18)*.

Menurut tafsir dari Kemenag, ayat ini mengingatkan setiap orang beriman untuk senantiasa bertakwa kepada Allah, serta memperhatikan perbuatannya sebagai bentuk persiapan menghadapi masa depan. Ayat ini juga mengandung pesan untuk introspeksi dan perencanaan yang matang berdasarkan pengetahuan dan kesadaran diri, agar langkah selanjutnya lebih baik dan bernilai (Chasbullah, 2020). Dalam konteks penelitian ini, penerapan prediksi pendapatan film berdasarkan data masa lalu mencerminkan makna dari ayat tersebut. Pendekatan ini mencerminkan ikhtiar dan perencanaan strategis yang sesuai dengan etika Islam, yaitu tidak gegabah dalam mengambil keputusan dan selalu mempertimbangkan dampak dari tindakan hari ini terhadap hasil di masa depan. Hasil prediksi ini dapat menjadi alat bantu pengambilan keputusan yang lebih bertanggung jawab, terencana, dan berlandaskan prinsip dalam Islam.

Dalam proses analisis dan prediksi, manusia menggunakan berbagai metode ilmiah untuk memperkirakan hasil yang akan datang. Metode yang diterapkan dalam penelitian ini adalah algoritma *Random Forest*, yang memanfaatkan data historis seperti anggaran produksi, pemain, dan jumlah penayangan *trailer* untuk

memprediksi pendapatan *box office* sebuah film. Prediksi ini berfungsi sebagai alat bantu dalam mengambil keputusan, tetapi pada akhirnya, keberhasilan atau kegagalannya tetap berada dalam ketentuan Allah. Sebagaimana dijelaskan dalam Al-Qur'an Surah Yusuf ayat 67, Allah berfirman:

وَقَالَ يٰبَنِيَّ لَا تَدْخُلُوْا مِنْۢ بَابٍ وَّاحِدٍ وَّادْخُلُوْا مِنْ اَبْوَابٍ مُّتَفَرِّقَةٍۗ وَمَآ اُغْنِيْ عَنْكُمْ مِّنَ اللّٰهِ مِنْ شَيْءٍۗ اِنِ الْحُكْمُ اِلَّا لِلّٰهِۗ عَلَيْهِ تَوَكَّلْتُ وَعَلَيْهِ فَلْيَتَوَكَّلِ الْمُتَوَكِّلُوْنَ ﴿٦٧﴾

"*Dia (Ya'qub) berkata, "Wahai anak-anakku, janganlah kamu masuk dari satu pintu gerbang, dan masuklah dari pintu-pintu gerbang yang berbeda-beda. (Namun,) aku tidak dapat mencegah (takdir) Allah dari kamu sedikit pun. (Penetapan) hukum itu hanyalah hak Allah. Kepada-Nyalah aku bertawakal dan hendaklah kepada-Nya (saja) orang-orang yang bertawakal (meningkatkan) tawakal(-nya)*" *(QS. Yusuf 12:67)*.

Menurut tafsir tahlili, ayat ini dijelaskan bahwa Nabi Ya'qub a.s. berkata kepada anak-anaknya agar ketika sampai di istana raja Mesir, mereka tidak masuk bersama-sama dari satu pintu gerbang, tetapi masuk dari pintu-pintu gerbang yang lain, supaya terhindar dari penglihatan mata orang yang hasad atau mengalami hal-hal yang tidak diinginkan. Di samping itu agar Bunyamin sempat bertemu dengan Yusuf secara terpisah dari saudara-saudaranya yang lain. Nabi Ya'qub a.s. menasihatkan pula bahwa walaupun mereka sudah berusaha menghindari berbagai kemungkinan yang membahayakan, namun beliau tidak dapat mencegah ketentuan dari Allah, sebab keputusan menetapkan sesuatu hanya berada di tangan-Nya. Semua pekerjaan harus dilaksanakan sesuai dengan kemampuan dan disertai keyakinan bahwa ketentuan dari Allah pasti terjadi, dan tidak seorang pun yang dapat menghalang-halanginya. Oleh karena itu, hanya kepada-Nya lah semua orang bertawakal dan berserah diri (Redaksi, 2021).

Dalam konteks penelitian ini, prediksi pendapatan *box office* film menggunakan algoritma *random forest* dapat dianalogikan sebagai bentuk ikhtiar manusia dalam menganalisis data dan mengambil keputusan. Namun, sebagaimana dijelaskan dalam ayat ini, meskipun strategi dan perhitungan telah dilakukan sebaik mungkin, hasil akhirnya tetap merupakan ketetapan Allah. Oleh karena itu, prediksi hanyalah alat bantu, sementara keberhasilan atau kegagalan suatu film di pasaran tetap bergantung pada banyak faktor yang berada di luar kendali manusia.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini dilakukan untuk memprediksi pendapatan *Box Office* Film menggunakan algoritma *Random Forest* dengan tiga variabel, yaitu anggaran produksi, ketenaran pemain, dan jumlah penayangan *trailer*. *Dataset* yang digunakan terdiri dari 500 data film. Evaluasi model dilakukan melalui dua pendekatan, yaitu pengujian berdasarkan rasio pembagian data *training* dan *testing*, serta validasi menggunakan K-*Fold Cross Validation*. Berdasarkan rasio pembagian data, pengujian pertama dilakukan dengan rasio 9:1, yang menghasilkan nilai MAE sebesar 0.36, MSE 0.32, dan R^2 0.6934. Pengujian ke-dua dengan rasio 8:2 memperoleh nilai MAE sebesar 0.41, MSE 0.37, dan R^2 0.6550. Pengujian ke-tiga dilakukan dengan rasio 7:3, dan didapatkan nilai MAE sebesar 0.40, MSE 0.34, dan R^2 0.6709. Dari ketiga skenario ini, rasio 9:1 menunjukkan performa terbaik dengan nilai kesalahan prediksi paling rendah dan nilai R^2 tertinggi. Hasil ini mengindikasikan bahwa proporsi data *training* yang lebih besar berpengaruh terhadap kualitas prediksi model. Pengujian K-*Fold Cross Validation* melibatkan dua skenario pengujian, yaitu 5-*Fold* dan 10-*Fold*. Pada 5-*Fold Cross Validation*, rata-rata nilai R^2 yang diperoleh adalah 0.6571, MAE sebesar 0.41, dan MSE sebesar 0.34. Sementara itu, pada skenario 10-*Fold Cross Validation*, rata-rata nilai R^2 adalah 0.6468, MAE sebesar 0.41, dan MSE 0.34. Hasil pengujian ini mengindikasikan bahwa model menunjukkan performa yang cukup baik pada

kedua skenario pengujian, meskipun masih ditemukan fluktuasi antar *fold*. Berdasarkan seluruh hasil pengujian, dapat disimpulkan bahwa algoritma *Random Forest* cukup baik digunakan untuk memprediksi pendapatan *box office* film berdasarkan variabel yang digunakan (Zachariah et al., 2014).

5.2 Saran

Berdasarkan hasil penelitian yang dilakukan dalam memprediksi pendapatan *box office* film menggunakan *Random Forest* berdasarkan variabel anggaran, pemain, dan jumlah penayangan *trailer*, peneliti menyadari dalam penelitian ini masih memiliki beberapa keterbatasan yang perlu diperhatikan. Saran-saran berikut dapat dipertimbangkan dalam pengembangan penelitian selanjutnya:

1. Mencoba dan membandingkan dengan algoritma prediksi lain seperti *Support Vector Regression* (SVR), *Gradient Boosting*, *XGBoost*, atau model *Deep Learning* sederhana. Saran ini dapat memberikan gambaran lebih menyeluruh terkait model mana yang paling optimal untuk memprediksi pendapatan film.
2. Menambahkan variabel pendukung yang relevan, seperti genre film, tanggal rilis, durasi film, rumah produksi, serta sutrada yang bertanggung jawab atas film tersebut.
3. Menggunakan *dataset* yang lebih besar dan bervariasi dari berbagai tahun rilis dan negara produksi. Saran ini dapat meningkatkan stabilitas model terhadap variasi karakteristik film.
4. Menghitung dan mengumpulkan pendapatan film satu tahun setelah tanggal rilis tiap film. Saran ini dapat memberikan hasil yang lebih adil dan seimbang antara film lama dan film baru.

DAFTAR PUSTAKA

Alqalyoobi, S. (2024). Prediction Model Development and Evaluation. *CHEST*, *165*(4), e131. https://doi.org/10.1016/j.chest.2023.11.045

Apriliah, W., Kurniawan, I., Baydhowi, M., & Haryati, T. (2021). Prediksi Kemungkinan Diabetes pada Tahap Awal Menggunakan Algoritma Klasifikasi Random Forest. *SISTEMASI*, *10*(1), 163. https://doi.org/10.32520/stmsi.v10i1.1129

Ardiansyah, F. (2024). MENGGUNAKAN ALGORITMA RANDOM FOREST UNTUK PREDIKSI HARGA PROPERTI. *Jurnal Dunia Data*, *1*(5), Article 5. http://www.pusdansi.org/index.php/duniadata/article/view/94

Ariatmanto, D., & Arief, M. I. (2023). PREDIKSI PELUANG KESUKSESAN FILM DALAM PRA PRODUKSI MENGGUNAKAN ALGORITMA DECISION TREE. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *7*(1), Article 1. https://doi.org/10.36040/jati.v7i1.6277

Arnaut, F., Kolarski, A., & Srećković, V. A. (2024). Machine Learning Classification Workflow and Datasets for Ionospheric VLF Data Exclusion. *Data*, *9*(1), Article 1. https://doi.org/10.3390/data9010017

Cabello-Solorzano, K., Ortigosa De Araujo, I., Peña, M., Correia, L., & J. Tallón-Ballesteros, A. (2023). The Impact of Data Normalization on the Accuracy of Machine Learning Algorithms: A Comparative Analysis. In P. García Bringas, H. Pérez García, F. J. Martínez De Pisón, F. Martínez Álvarez, A. Troncoso Lora, Á. Herrero, J. L. Calvo Rolle, H. Quintián, & E. Corchado (Eds.), *18th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2023)* (Vol. 750, pp. 344–353). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-42536-3_33

Celine, S., Dominic, M. M., Devi, M. S., Assistant Professor, Department of Computer Science, Sacred Heart College, Tirupattur, Tamil Nadu, India., Assistant Professor, Department of Computer Science, Periyar University Constitution College for Arts and Science, Harur, Tamil Nadu, India., & Research Scholar, Department of Computer Science, Sacred Heart College, Tirupattur, Tamil Nadu, India. (2020). Logistic Regression for Employability Prediction. *International Journal of Innovative Technology and Exploring Engineering*, *9*(3), 2471–2478. https://doi.org/10.35940/ijitee.C8170.019320

Chasbullah, A. (2020, July 27). Tafsir Surat Al-Hasyr Ayat 18: Introspeksi Diri, Manejemen Waktu. *Tafsir Al Quran | Referensi Tafsir di Indonesia*. https://tafsiralquran.id/tafsir-surat-al-hasyr-ayat-18-intropeksi-diri-manajemen-waktu-dan-tabungan-kebaikan-dalam-al-quran/

Cohen, S. (2021). Dealing with data: Strategies of preprocessing data. In *Artificial Intelligence and Deep Learning in Pathology* (pp. 77–92). Elsevier. https://doi.org/10.1016/B978-0-323-67538-3.00005-1

Dallah, D., & Sulieman, H. (2024). Outlier Detection Using the Range Distribution. In F. Kamalov, R. Sivaraj, & H.-H. Leung (Eds.), *Advances in Mathematical Modeling and Scientific Computing* (pp. 687–697). Springer International Publishing. https://doi.org/10.1007/978-3-031-41420-6_57

Dwiyanti, Z. A., & Prianto, C. (2023). Prediksi Cuaca Kota Jakarta Menggunakan Metode Random Forest. *Jurnal Tekno Insentif*, *17*(2), Article 2. https://doi.org/10.36787/jti.v17i2.1136

Fadlilah, A. D. N., Wahyuningsih, Y., & Khawarga, Y. A. (2023). Prediksi Spesies Burung Menggunakan Random Forest. *KOMIK (Konferensi Nasional Teknologi Informasi Dan Komputer)*, *6*(1), Article 1. https://doi.org/10.30865/komik.v6i1.5783

Geng, X., Ma, Y., Cai, W., Zha, Y., Zhang, T., Zhang, H., Yang, C., Yin, F., & Shui, T. (2023). Evaluation of models for multi-step forecasting of hand, foot and mouth disease using multi-input multi-output: A case study of Chengdu, China. *PLOS Neglected Tropical Diseases*, *17*(9), e0011587. https://doi.org/10.1371/journal.pntd.0011587

Gruber, S. G., & Bach, F. (2024). *Optimizing Estimators of Squared Calibration Errors in Classification* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2410.07014

Hadi, A. S., & Imon, A. H. M. R. (2024). Some Recent Developments in the Identification of Outliers in Spatial Data and Spatial Regression. In M. Masoom Ali, R. Imon, I. Ali, & H. M. Yousof, *Statistical Outliers and Related Topics* (1st ed., pp. 19–35). CRC Press. https://doi.org/10.1201/9781003379881-2

Hadi, N., & Benedict, J. (2024). Implementasi Machine Learning Untuk Prediksi Harga Rumah Menggunakan Algoritma Random Forest. *Computatio : Journal of Computer Science and Information Systems*, *8*(1), Article 1. https://doi.org/10.24912/computatio.v8i1.15173

Haidar, D., Irawan, B., & Bahtiar, A. (2024). PENERAPAN DEEP LEARNING MODEL RANDOM FOREST UNTUK PREDIKSI PENERIMA BANTUAN PROGRAM KELUARGA HARAPAN (PKH). *JATI (Jurnal Mahasiswa Teknik Informatika)*, *7*(6), 3564–3571. https://doi.org/10.36040/jati.v7i6.8250

Hao, B. (2023). The Analysis of the Factors that Influence the Film Revenue. *Highlights in Science, Engineering and Technology*, *47*, 154–159. https://doi.org/10.54097/hset.v47i.8184

Hodson, T. O. (2022). Root-mean-square error (RMSE) or mean absolute error (MAE): When to use them or not. *Geoscientific Model Development*, *15*(14), 5481–5487. https://doi.org/10.5194/gmd-15-5481-2022

Horváth, Á., & Gyenge, B. (2023). Changes and trends based on perceived lifestyles reflected in movies. *SocioEconomic Challenges*, *7*(3), 174–183. https://doi.org/10.61093/sec.7(3).174-183.2023

Huang, S. (2024). Film marketing strategy analysis on social media—A case study of the film "YOLO." *SHS Web of Conferences*, *199*, 03020. https://doi.org/10.1051/shsconf/202419903020

Jackins, V., Vimal, S., Kaliappan, M., & Lee, M. Y. (2021). AI-based smart prediction of clinical disease using random forest classifier and Naive Bayes. *The Journal of Supercomputing*, *77*(5), 5198–5219. https://doi.org/10.1007/s11227-020-03481-x

JpnMuslim. (2015). *Tafsir Ibnu Katsir Lengkap pdf*. http://archive.org/details/Tafsir_Ibnu_Katsir_Lengkap_114Juz

Kamalov, F., Moussa, S., & Reyes, J. A. (2023). Data Transformation in Machine Learning: Empirical Analysis. *2023 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, 115–120. https://doi.org/10.1109/3ICT60104.2023.10391512

Kampani, J., & Nicolaides, C. (2023). Information consistency as response to pre-launch advertising communications: The case of YouTube trailers. *Frontiers in Communication*, *7*, 1022139. https://doi.org/10.3389/fcomm.2022.1022139

Kumar, G., Pandey, S. K., Varshney, N., Mishra, P., Singh, K. U., & Verma, K. (2023). Predicting the Success of Motion Pictures using Deep Learning. *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*, 1–8. https://doi.org/10.1109/ISCON57294.2023.10112202

Lase, Y., Claudia, S., Lingga, I., & Ariyudha, D. (2024). Uncovering the Secrets of the 2023 Box Office: Analysis of Factors Affecting the Success of the Top 200 Films Using the Linear Regression Method. *Electronic Integrated Computer Algorithm Journal*, *2*(1), 29–39. https://doi.org/10.62123/enigma.v2i1.45

Lee, T.-H., Ullah, A., & Wang, R. (2020). Bootstrap Aggregating and Random Forest. In P. Fuleky (Ed.), *Macroeconomic Forecasting in the Era of Big Data* (Vol. 52, pp. 389–429). Springer International Publishing. https://doi.org/10.1007/978-3-030-31150-6_13

Liu, Z. (2023). The Influence of Film Marketing Strategy on the Film Box Office. *Highlights in Business, Economics and Management*, *19*, 1–5. https://doi.org/10.54097/hbem.v19i.11744

Madongo, C. T., Tang, Z., & Hassan, J. (2023). Box-office Revenue Prediction by Mining Deep Features from Movie Posters and Reviews Using Transformers. *2023 6th International Conference on Artificial Intelligence and Pattern Recognition (AIPR)*, 1406–1412. https://doi.org/10.1145/3641584.3641796

Mahmud Sujon, K., Binti Hassan, R., Tusnia Towshi, Z., Othman, M. A., Abdus Samad, M., & Choi, K. (2024). When to Use Standardization and Normalization: Empirical Evidence From Machine Learning Models and XAI. *IEEE Access*, *12*, 135300–135314. https://doi.org/10.1109/ACCESS.2024.3462434

Montgomery, R. M. (2024). *Techniques for Outlier Detection: A Comprehensive View*. https://doi.org/10.20944/preprints202410.1603.v1

Moretti, A., Shlomo, N., & Sakshaug, J. W. (2020). Parametric Bootstrap Mean Squared Error of A Small Area Multivariate EBLUP. *Communications in Statistics - Simulation and Computation*, *49*(6), 1474–1486. https://doi.org/10.1080/03610918.2018.1498889

Mushaf Al-Qur'an, L. P. (2016, November). Tafsir Al-Quran Kemenag. *Terjemah Kitab Kuning*. https://www.alkhoirot.org/2024/06/tafsir-al-quran-kemenag.html

Musthofa, K. (2021, August 8). Belajar Investasi dari Nabi Yusuf, Tafsir Surah Yusuf Ayat 47-49. *Tafsir Al Quran | Referensi Tafsir di Indonesia*. https://tafsiralquran.id/belajar-investasi-dari-nabi-yusuf-tafsir-surah-yusuf-ayat-47-49/

Novrian, R., Agustiani, T., Fikri, M., Hikmatulloh, M. F., Gunawan, M. E., & Firdaus, U. (2024). Penerapan Algoritma Random Forest dalam Prediksi Status Penerima PIP pada Siswa: Studi Kasus pada SMK Amaliah 1. *Karimah Tauhid*, *3*(2), Article 2. https://doi.org/10.30997/karimahtauhid.v3i2.11937

Pospisil, D. A., & Bair, W. (2021). The unbiased estimation of the fraction of variance explained by a model. *PLOS Computational Biology*, *17*(8), e1009212. https://doi.org/10.1371/journal.pcbi.1009212

Rais, A. N., Warjiyono, W., Alfarobi, I., Hadi, S. W., & Kurniawan, W. (2024). ANALISA PREDIKSI HARGA JUAL RUMAH MENGGUNAKAN ALGORITMA RANDOM FOREST MACHINE LEARNING. *Jurnal Riset Sistem Informasi Dan Teknologi Informasi (JURSISTEKNI)*, *6*(2), Article 2. https://doi.org/10.52005/jursistekni.v6i2.323

Rather, A. A., Kinjawadekar, R., Ahmad, A., Dar, A. M., Nisa, R., & Elemary, B. R. (2024). Statistical Outliers and a New Generalized Probability Distribution with Applications. In M. Masoom Ali, R. Imon, I. Ali, & H. M. Yousof, *Statistical Outliers and Related Topics* (1st ed., pp. 135–152). CRC Press. https://doi.org/10.1201/9781003379881-7

Redaksi. (2021, April 27). Tafsir Surah Yusuf ayat 67-77, Perjumpaan Yusuf dan Bunyamin. *Tafsir Al Quran | Referensi Tafsir di Indonesia*. https://tafsiralquran.id/tafsir-surah-yusuf-ayat-67-77/

Robeson, S. M., & Willmott, C. J. (2023). Decomposition of the mean absolute error (MAE) into systematic and unsystematic components. *PLOS ONE*, *18*(2), e0279774. https://doi.org/10.1371/journal.pone.0279774

Saadah, S., & Salsabila, H. (2021). Prediksi Harga Bitcoin Menggunakan Metode Random Forest:(Studi Kasus: Data Acak Pada Masa Pandemic Covid-19). *Jurnal Komputer Terapan*, *7*(1), 24–32.

Sandag, G. A. (2020). Prediksi Rating Aplikasi App Store Menggunakan Algoritma Random Forest. *CogITo Smart Journal*, *6*(2), 167–178. https://doi.org/10.31154/cogito.v6i2.270.167-178

Satya Sree, K. P. N. V., Karthik, J., Niharika, C., Srinivas, P. V. V. S., Ravinder, N., & Prasad, C. (2021). Optimized Conversion of Categorical and Numerical Features in Machine Learning Models. *2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 294–299. https://doi.org/10.1109/I-SMAC52330.2021.9640967

Schonlau, M. (2023). Random Forests. In M. Schonlau, *Applied Statistical Learning* (pp. 183–204). Springer International Publishing. https://doi.org/10.1007/978-3-031-33390-3_10

Singh, K., Rokde, J., & Simran. (2024). Film Box Office Success Forecasting: A Genre-Driven Classification System Using Machine Learning and Cluster Analysis. *2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, 486–492. https://doi.org/10.1109/ICAAIC60222.2024.10574956

Suhartono, S. (2013). Pemodelan Pertumbuhan Tanaman Zinnia Menggunakan Lindenmayer System dengan Mathematica. *CAUCHY: Jurnal Matematika Murni Dan Aplikasi*, *3*(1), 33–37. https://doi.org/10.18860/ca.v3i1.2569

Talaei Khoei, T., & Kaabouch, N. (2023). Machine Learning: Models, Challenges, and Research Directions. *Future Internet*, *15*(10), Article 10. https://doi.org/10.3390/fi15100332

Tang, L. (2022). *Analysis of the Factors that Make People Love* Get Out *and* 1917*: Why the Two Films with Such Different Production Budget Can Gain Such Similar Popularity:* 2022 International Conference on Comprehensive Art and Cultural Communication (CACC 2022), Chongqing, China. https://doi.org/10.2991/assehr.k.220502.064

Tofallis, C. (2021). A Better Measure of Relative Prediction Accuracy for Model Selection and Model Estimation. *Journal of the Operational Research Society*, *66*(8), 1352–1362. https://doi.org/10.1057/jors.2014.103

Tsiligaridis, J. (2023). Tree-Based Ensemble Models, Algorithms and Performance Measures for Classification. *Advances in Science, Technology and Engineering Systems Journal*, *8*(6), 19–25. https://doi.org/10.25046/aj080603

Udupi, P. K., Dattana, V., Netravathi, P. S., & Pandey, J. (2023). Predicting Global Ranking of Universities Across the World Using Machine Learning Regression Technique. *SHS Web of Conferences*, *156*, 04001. https://doi.org/10.1051/shsconf/202315604001

Vidya Chitre. (2024). Exploring Machine Learning Techniques for Predictive Analytics in Computational Mathematics. *Panamerican Mathematical Journal*, *34*(2), 1–19. https://doi.org/10.52783/pmj.v34.i2.919

Wahyudi, B. (2022). Prediksi Peringkat Aplikasi di Google Play Menggunakan Metode Random Forest. *Jurnal Nasional Teknologi Komputer*, *2*(1), 38–47. https://doi.org/10.61306/jnastek.v2i1.25

Wu, M. (2024). Analysing the Global Marketing Strategies of Major Film and Television Studios: A Case Study Based on Marvel. *Highlights in Business, Economics and Management*, *41*, 166–170. https://doi.org/10.54097/czw8nf95

Xie, C. (2024). A refined approach to early movie box office prediction leveraging ensemble learning and feature encoding. *Applied and Computational Engineering*, *75*(1), 273–284. https://doi.org/10.54254/2755-2721/75/20240555

Zachariah, N., Kothari, S., Ramamurthy, S., Osunkoya, A. O., & Wang, M. D. (2014). Evaluation of performance metrics for histopathological image classifier optimization. *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 1933–1936. https://doi.org/10.1109/EMBC.2014.6943990

Zhao, Z., Chuah, J. H., Lai, K. W., Chow, C.-O., Gochoo, M., Dhanalakshmi, S., Wang, N., Bao, W., & Wu, X. (2023). Conventional machine learning and deep learning in Alzheimer's disease diagnosis using neuroimaging: A review. *Frontiers in Computational Neuroscience*, *17*. https://doi.org/10.3389/fncom.2023.1038636