## Penerapan Algoritma Fuzzy untuk Memberikan Saran yang Optimal dalam Pengambilan Keputusan pada Permainan Kartu Monster

**Muhammad Rizky Perdana\*1, Andika Sundawijaya2**

1,2Program Studi Ilmu Komputer; STIMIK ESQ Business School

Jl. Tb. Simatupang, Kav. 1 Cilandak Timur, Jakarta 11560

e-mail: [\*1m.rizky.p@students.esqbs.ac.id](mailto:*1m.rizky.p@students.esqbs.ac.id), 2 [sundawijaya@esqbs.ac.id](mailto:sundawijaya@esqbs.ac.id)

***Abstrak:***Kecerdasan buatan merupakan bahasan yang cukup diminati oleh banyak orang. Adapun untuk dunia permainan*,* kecerdasan buatan mampu memberikan inovasi yang baik terhadap permainan yang menggunakannya. Sebagai lawan, kecerdasan buatan mampu memberikan tantangan dengan intelektualitas tertentu dalam berpikir, sehingga pemain tidak harus mencari lawan tanding untuk bermain. Permainan strategi adalah salah satu genre permainan yang mana pemain harus memikirkan cara dengan menentukan pemikiran dan perencanaan yang matang dalam memperoleh kemenangan*.* Permainan kartu merupakan salah satu dari genre permainan strategi tersebut yang tepat jika menggunakan kecerdasan buatan sebagai lawan bermain atau pemberi saran pemilihan langkah yang tepat dalam permainan, Pada tipe permainan ini pemain bermain bergiliran dan memiliki kesempatan mengatur rencana dalam menjatuhkan lawannya. Penerapan kecerdasan buatan pada permainan kartu sebagai pemberi saran pemilihan langkah, menjadi dasar dilaksanakannya penelitian ini. Pada penelitian ini akan menerapkan algoritma *fuzzy* dengan inferensi Mamdani dalam menentukan saran pemilihan kartu yang optimal pada permainan kartu Yu-Gi-Oh. Pemilihan saran kartu menggunakan algoritma fuzzy Mamdani dilakukan berulang kali sebanyak 15 kali menggunakan data kartu Yu-Gi-Oh yang sudah diolah. Hasil dari model yang diperoleh yaitu akurasi pemilihan saran kartu yang optimal sebesar 0,7333 yang menunjukan bahwa akurasi penggunaan sistem yang sudah dirancang adalah cukup baik. Dari hasil tersebut peneliti memberikan rekomendasi dengan menambah kartu pada dataset seperti beberapa tipe kartu yang berbeda dan menambahkan atau menggunakan metode yang berbeda serta Analisis penambahan input pada variabel *fuzzy* untuk menambah keakuratan sistem dalam memilih saran kartu yang lebih optimal.

***Kata Kunci:*** *Permainan Kartu Monster, Fuzzy Mamdani, Kecerdasan Buatan, Pemberi Saran Permainan, Ilmu Komputer.*

# pendahuluan

Kecerdasan buatan merupakan bahasan yang cukup diminati oleh banyak orang. Adapun untuk dunia permainan, kecerdasan buatan mampu memberikan inovasi yang baik terhadap permainan yang menggunakannya. Sebagai lawan, kecerdasan buatan mampu memberikan tantangan dengan intelektualitas tertentu dalam berpikir, sehingga pemain tidak harus mencari lawan tanding untuk bermain. Kecerdasan buatan mampu memberikan keputusan yang tepat didasari oleh pemilihan algoritma yang efektif dan efisien sesuai dari kondisinya untuk dunia game (Uari dkk., 2021). Berdasarkan pemahaman tersebut, menjadikan Kecerdasan buatan banyak diterapkan pada banyak genre dan tipe permainan, baik pada permainan komputer atau bahkan simulasi dari beberapa permainan di dunia nyata.

Permainan strategi adalah salah satu genre permainan yang mana pemain harus memikirkan cara dengan menentukan pemikiran dan perencanaan yang matang dalam memperoleh kemenangan (Subiantoro dkk., 2019). Permainan kartu monster Yu-Gi-Oh merupakan salah satu dari genre permainan strategi kartu yang dimainkan oleh dua orang pemain atau lebih, yang disesuaikan dengan peraturan permainannya. Banyak jenis dan variasi tipe kartu pada permainan tersebut serta beberapa cara pemanggilan dalam memainkan kartu monster adalah nilai tambah pada permainan kartu Yu-Gi-Oh. Dalam permainannya pemain menarik dari kumpulan kartu masing-masing pemain sebanyak 5 kartu dan akan menarik sebanyak 1 kartu setiap gilirannya, pemain hanya bisa mengeluarkan 1 kartu monster dan beberapa kartu pendukung serta jebakan di setiap giliran. Pada tipe permainan ini pemain bermain bergiliran dan memiliki kesempatan mengatur rencana dalam menjatuhkan lawannya. Setiap pergerakan dapat mempengaruhi alur dari setiap permainan, sehingga kesalahan dalam pengambilan langkah akan berakibat fatal (Subiantoro dkk., 2019). Pemain akan saling mengurangi point kehidupan lawannya, mereka memiliki poin kehidupan sebesar 4000 atau 8000 yang perlu dijaga agar tidak menyentuh angka 0 yang berarti kalah dan pemain yang mampu bertahan setelah pemain lain menyentuh angka 0 akan menang.

Permainan kartu monster Yu-Gi-Oh memiliki alur permainan yang sederhana namun cukup rumit. Sederhana karena permainan dilakukan bergiliran dengan mengeluarkan kartu yang tepat untuk menyerang atau bertahan dengan tujuan mengurangi poin kehidupan musuh dan melindungi poin kehidupan pemain sendiri, serta rumit karena banyak tipe, nama dan jenis kartu serta besar atau kecil poin serangan dan bertahan kartu yang ada pada permainan tersebut. Menentukan kartu yang tepat untuk digunakan dalam permainan juga memerlukan informasi yang tepat pada suatu kondisi permainan. Banyak pemain salah mengambil langkah dalam permainan, yang mampu menyebabkan kesempatan kalah lebih besar pada permainan tersebut.

Untuk mengatasi masalah tersebut diperlukan sebuah sistem saran yang mampu membantu pemain dalam menentukan kartu yang sesuai untuk digunakan dalam permainan tersebut. Sistem saran akan memberikan masukan kepada pemain dalam menentukan kartu yang tepat dan optimal dalam suatu langkah, sehingga akan membantu pemain memilih kartu yang tepat dalam permainan.

Dalam memberikan saran, sistem memerlukan sebuah algoritma yang mampu memetakan beberapa kondisi serta masukan dan memberikan keputusan. Dalam permainan kartu Yu-Gi-Oh sistem saran akan dibutuhkan melihat dari kondisi permainan demi mendapatkan nilai output yang akurat. Ada beberapa algoritma yang mampu memetakan dan menentukan keputusan yang bisa menjadi sebuah sistem saran, namun melihat pada permainan kartu Yu-Gi-Oh memiliki banyak ketentuan nilai yang ambigu dalam permainan, membuat algoritma fuzzy yaitu teori himpunan logika yang dikembangkan untuk mengatasi konsep nilai yang terdapat diantara kebenaran dan kesalahan (Raharjo dkk., 2020) menjadi algoritma yang tepat jika digunakan pada permainan tersebut.

Pada penelitian ini, peneliti akan menggunakan algoritma yang sering digunakan dalam sebuah permainan yaitu logika fuzzy pengoptimalan dari logika boolean sebagai basis pengetahuan pada sistem saran permainan. Hal ini dilakukan karena dalam permainan kartu monster Yu-Gi-Oh memiliki nilai keambiguan seperti kuat atau lemah, sehingga logika fuzzy tepat digunakan karena memiliki nilai keanggotaan antara 0 dan 1 dan nilai linguistik seperti konsep “sedikit”, “lumayan” dan “sangat” (Raharjo dkk., 2020). Mengacu pada penjelasan sebelumnya, sistem saran sangat memungkinkan dibuat menggunakan algoritma fuzzy, namun sistem tersebut mampu memberikan nilai yang akurat atau tidak masih belum diketahui. Dalam penelitian ini akan menunjukan seberapa akurat sistem saran menggunakan algoritma fuzzy tersebut dalam permainan kartu monster Yu-Gi-Oh.

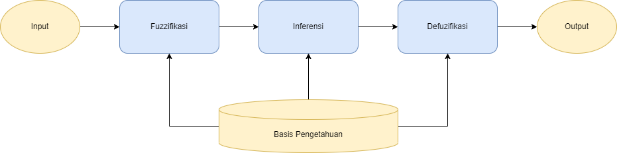
Penelitian ini diharapkan dapat menerapkan logika fuzzy Mamdani dalam menentukan strategi saran kartu yang tepat dan optimal dalam permainan sehingga mampu mengurangi kesalahan pemain dalam menentukan kartu dalam permainan.

# landasan teori

## *Fuzzy*

Algoritma Fuzzy merupakan suatu metode yang berguna untuk memecahkan masalah ketidakpastian atau memiliki ambiguitas. Teori ini menyatakan bahwa derajat keanggotaan dari suatu elemen bukan hanya terdiri dari 0 dan 1, melainkan dari rentang 0 sampai 1 (Nurdiyanto dkk., 2017). Metode Fuzzy juga merupakan salah satu metode yang sering digunakan dalam sistem pengambil keputusan. Logika fuzzy adalah logika multivalued, yang memungkinkan nilai menengah harus didefinisikan antara evaluasi konvensional seperti benar atau salah, ya atau tidak dan tinggi atau rendah (Tanjung dkk., 2019).

Secara garis besar, algoritma logika fuzzy meliputi fuzzification, rule evaluation atau inferensi dan defuzzification. Pada fuzzification dirancang variable sebagai penentu dalam fungsi keanggotaan dan mengubah nilai tegas menjadi nilai fuzzy yang berada dalam fungsi keanggotaan, rule evaluation dan Inferensi adalah aturan “jika-maka” yang digunakan sebagai basis pengetahuan sistem dalam memperoses data dan defuzzification yang mengubah nilai fuzzy yang sudah diproses menjadi nilai tegas kembali sehingga menjadi nilai output dari proses tersebut. Adapun detail dari sistem inferensi fuzzy dapat dilihat pada Gambar 2.1



Gambar 1. SIstem Inferensi *Fuzzy*

### ***Fuzzy* Mamdani**

Fuzzy Mamdani merupakan salah satu dari beberapa metode yang ada pada logika fuzzy. Metode mamdani sering digunakan dalam aplikasi-aplikasi karena strukturnya yang sederhana, yaitu menggunakan operasi Min-Max. Untuk mendapatkan output diperlukan empat tahapan yaitu, Pembentukan himpunan fuzzy, Aplikasi fungsi implikasi, Komposisi aturan dan defuzifikasi (Wanto, 2019). Didalam metode Mamdani proses implikasi fuzzy menggunakan operator min atau conjunction yang menandakan nilai minimum, sedangkan aturan agregasi menggunakan operator max yang merupakan nilai maksimumnya (Raharjo dkk., 2020).

## Yu-Gi-Oh

Yu-Gi-Oh adalah sebuah manga (komik jepang) yang diciptakan oleh Kazuki Takahashi. Perkembangan Yu-Gi-Oh sudah banyak menjadi waralaba meliputi acara televisi, tanding kartu, video game dan anime (film seri atau movie) pada zaman mesir kuno yang senantiasa memainkan suatu permainan kartu. Permainan kartu Yu-Gi-Oh merupakan permainan strategi yang menarik dan asik dimainkan segala kalangan. Secara garis besar permainan ini terdapat tiga macam kartu, yaitu kartu Monster, Spell (Sihir) dan Trap (Jebakan) (Firdaus & Puspasari, 2020)*.*

Permainan dimulai dengan kedua pemain menarik lima kartu dari kumpulan kartu atau biasa disebut dengan Deck. Pemain mulai menjalankan permainan tersebut secara bergiliran dengan mengeluarkan kartu monster mereka ke papan pertandingan yang memiliki tujuan untuk mengurangi poin kehidupan lain dalam permainan tersebut. Kedua pemain memiliki 8000 poin atau 4000 poin kehidupan tergantung dari jenis aturannya yang perlu dijaga, agar tidak sampai kepada poin 0 yang menunjukan pemain tersebut kalah. dalam kondisi tersebut kedua pemain bisa mengatur strategi terbaik bagi mereka dalam mengalahkan musuh-musuhnya dengan cara menentukan kartu yang tepat untuk digunakan dalam mengalahkan lawanya.

### **Normal Monster**

Normal monster merupakan kartu monster yang memiliki poin attack, poin defense, level bintang dan tidak memiliki effect apapun pada kartu tersebut. Kartu normal monster memiliki poin terkecil 0 poin dan poin terbesar 3000 point pada attack dan defense. Contoh kartu seperti pada Gambar 2.2.



Gambar 1. Normal Monster *Dark Magician*

# metodologi penelitian

## pengambilan dan pengumpulan data

Data yang akan diolah pada penelitian ini adalah data sekunder. Data yang digunakan adalah pengumpulan sampel kecil dari beberapa kartu yang ada pada kumpulan data kartu Yu-Gi-Oh Kaggle yang bersumber dari data API pada ygoprodeck https://www.kaggle.com/datasets/ioexception/yugioh-cards. Adapun untuk pengujian akan tetap menggunakan kartu dari sampel kecil tersebut dengan ketentuan sebagai berikut :

* Jumlah kartu pada sistem adalah maksimal 5 pada papan pemain, 5 pada papan musuh dan 6 pada tangan pemain.
* Jumlah pemain hanya 2 orang.
* Kartu yang digunakan adalah kartu *normal monster*.

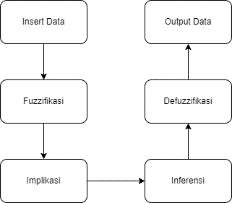
Teknik pengumpulan data bersifat Kuantitatif dengan cara mencari beberapa kartu yang peneliti telah tentukan sebagai subjek penelitian. Data bersumber dari Kaggle yang bersumber dari API ygoprodeck dan akan diambil sampel serta dikumpulkan sesuai dengan kriteria yang diperlukan.

## Analisis Data

Kartu YuGiOh memiliki banyak sekali variabel, seperti *attack, defence, level, type, attribute,* dan *effect.* Data dari setiap kartu akan dilakukan filtrasi sehingga data yang digunakan sesuai dengan sampel data yang dibutuhkan. Kemudian data akan dilakukan fuzzifikasi dan inferensi menggunakan aturan *fuzzy* yang berguna untuk menjadi acuan sistem dalam menentukan saran optimal dalam pemilihan kartu serta defuzzifikasi untuk mendapatkan output tegas dari hasil fuzzifikasi dan inferensi yang sudah dilakukan.

## Pengujian Data

Pada pengujian data, data yang sudah dipilih sebagai sampel akan dilakukan fuzzifikasi sehingga data kartu yang ada bisa menghasilkan kombinasi aturan. Setelah terbentuk aturan, data akan masuk kedalam proses implikasi yang akan menghasilkan keluaran-keluaran berupa himpunan fuzzy yang mana himpunan tersebut akan digabungkan dengan himpunan-himpunan lainnya untuk digunakan pada proses defuzzifikasi. Dalam proses inferensi mamdani nilai dari himpunan-himpunan tersebut peneliti akan menggunakan fungsi OR, yang mana pada fungsi ini digunakan untuk mendapatkan nilai maksimum dari perbandingan himpunan yang memiliki nilai yang sama. Pada proses defuzzifikasi data yang maksimum akan diubah menjadi nilai yang tegas. Dan terakhir nilai yang dihasilkan pada kombinasi aturan akan dimasukan kedalam proses defuzzifikasi. Eksperimen pengujian data bisa dilihat pada Gambar 3.2.



Gambar 5. Alur Pengujian Data

Pada pengujian sistem, akurasi akan dihitung melalui total keseluruhan uji coba yang berhasil dibagi dengan total keseluruhan uji coba sehingga akan menghasilkan nilai akurasi juga tingkat error suatu sistem. Adapun formula perhitungan akurasi adalah sebagai berikut :

ΣA/ΣB \* 100%

Dimana ΣA adalah jumlah jumlah seluruh percobaan yang berhasil dan ΣB adalah seluruh percobaan yang telah dilakukan.

## Interpretasi Data

Pada tahap ini, peneliti akan mempelajari lebih lanjut hasil data yang sudah diolah. Hasil tersebut akan ditinjau ulang dengan analisis dan teori dari studi literatur sebelumnya. Setelah aturan *fuzzy* sudah ada maka data tersebut baru bisa di proses dalam sistem inferensi *fuzzy* Mamdani sehingga menghasilkan *output*, yang pada *output* tersebut dapat peneliti lihat seberapa tepat aturan *fuzzy* yang digunakan tersebut untuk menentukan saran pemilihan kartu yang optimal pada sistem dalam pemilihan kartu yang tepat pada permainan YuGiOh tersebut. Selain itu, pengujian berkala juga dilakukan pada sistem untuk memastikan apakah sistem tersebut membuahkan hasil yang benar atau tidak dengan cara menentukan nilai akurasi dari hasil pengujian berkala tersebut. Setelah semua dilakukan peneliti akan menuliskan kesimpulan dan saran dari interpretasi hasil pengujian sistem pada permainan tersebut.

# hasil dan analisis

## Analisis

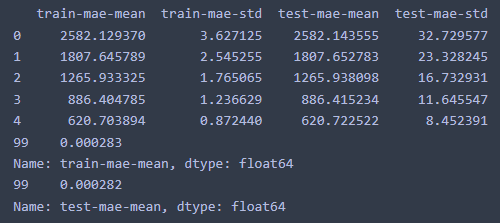
Pada penelitian ini peneliti mengambil himpunan data yang berasal dari website Kaggle Antonio Feregrino Bolaños yang terakhir diunggah pada tahun 2021. Adapun attribute yang ada pada data tersebut terdapat 18 kolom, yaitu id, name, type, desc, atk, def, level, race, attribute, scale, archetype, linkval, linkmarkers, img\_url, img\_url\_small, ban\_tcg, ban\_ocg, ban\_goat. Himpunan data tersebut dapat diunduh pada link https://www.kaggle.com/datasets/ioexception/yugioh-cards. Dikarenakan Terdapat banyaknya kolom atribut yang kurang berguna dalam penelitian, maka perlu bagi peneliti untuk mengambil beberapa atribut kolom yang diperlukan seperti id sebagai nilai indeks yang membedakan setiap kartu, name sebagai nama dari setiap kartu, type jenis dari kartu yang ada, atk adalah nilai serangan suatu kartu, def nilai bertahan suatu kartu dan level adalah nilai level suatu kartu yang menentukan kartu tersebut bisa diaktifkan atau tidak pada himpunan data tersebut. Rincian data pada himpunan data kartu tersebut dapa dilihat pada Tabel 4.1.

Tabel 10. Parameter Pengujian Model Setiap Skenario

|  |  |  |
| --- | --- | --- |
| Parameter | Skenario Uji Model 1 | Skenario Uji Model 2 |
| *Default Parameter Values* | *GridSearchCV Parameters Values* |
| n\_estimators | 100 | 1000 |
| max\_depth | 6 | 3 |
| learning\_rate | 0,3 | 0,05 |
| reg\_lambda | 1 | 1 |
| reg\_alpha | 0 | 1 |
| gamma | 0 | 0 |
| min\_child\_weight | 1 | 0,5 |
| early\_stopping\_round | *None* | 10 |
| n\_estimators | 100 | 1000 |
| max\_depth | 6 | 3 |
| learning\_rate | 0,3 | 0,05 |

Pada skenario pengujian model satu, telah dilakukan pembuatan model yang menggunakan parameter bawaan *XGBoost*. Hasil dari skenario tersebut lalu dievaluasi untuk mengecek seberapa baik model tersebut. Evaluasi kedua skenario uji model tersebut dilakukan menggunakan *Mean Absolute Error* (MAE) dan R Square ().

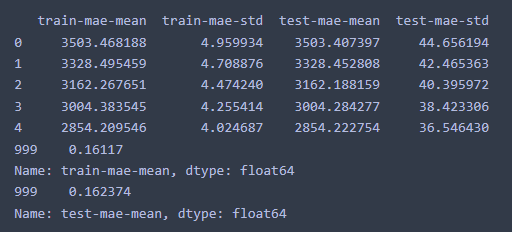
Hasil evaluasi nilai MAE pada skenario uji model satu untuk data latih dan data uji masing-masing bernilai 0,000295 dan 0,000293. Sedangkan evaluasi menggunakan pada data latih dan data uji masing-masing menunjukkan nilai yang sama yaitu 0,99. Dapat dilihat jika pemodelan dilakukan tanpa menyetel parameter menghasilkan nilai error yang sangat rendah. Untuk mengecek apakah model *overfitting*, *underfitting*, atau baik dilakukan *cross validation* dengan jumlah *fold* sebanyak 10 dan iterasi sebanyak 100. Dari validasi tersebut didapatkan hasil error untuk data latih dan data uji masing-masing bernilai 0,000283 dan 0,000282 pada iterasi ke-99 seperti ditunjukkan pada Gambar 6.



Gambar 6. Hasil *Cross Validation* Skenario Uji Model 1

Berdasarkan Gambar 6, nilai error dari data latih dan data uji yang dihasilkan *cross validation* terdapat selisih sebesar 0,000012 dan 0,000011. Makna dari nilai menggunakan *cross validation* menujukkan bahwa model dapat dikatakan cukup baik.

Pada skenario uji model dua, hasil evaluasi nilai MAE yaitu sebesar 0,1353 dan 0,1308 untuk masing-masing data latih dan data uji. Sedangkan evaluasi menggunakan pada data latih dan data uji masing-masing juga menunjukkan nilai yang sama yaitu 0,99. Nilai error yang dihasilkan sedikit lebih tinggi daripada skenario uji model pertama, pada data latih selisih nya sebesar 0,135005 dan data uji sebesar 0,130507. Validasi juga dilakukan untuk melihat baik tidak nya model dengan *fold* berjumlah 10 dan iterasi sebanyak 1000. Dari validasi tersebut didapatkan hasil error untuk data latih dan data uji masing-masing bernilai 0,16117 dan 0,162374 pada iterasi ke-999 seperti ditunjukkan pada Gambar 7.



Gambar 7. Hasil *Cross Validation* Skenario Uji Model 2

Berdasarkan Gambar 7, nilai error dari data latih dan data uji yang dihasilkan *cross validation* terdapat selisih sebesar 0,02587 dan 0,031574. Makna dari nilai menggunakan *cross validation* menujukkan bahwa model dapat dikatakan sangat baik. Hal ini dikarenakan nilai validasi error untuk data latih dan data uji sedikit lebih besar daripada nilai error untuk data latih dan data uji.

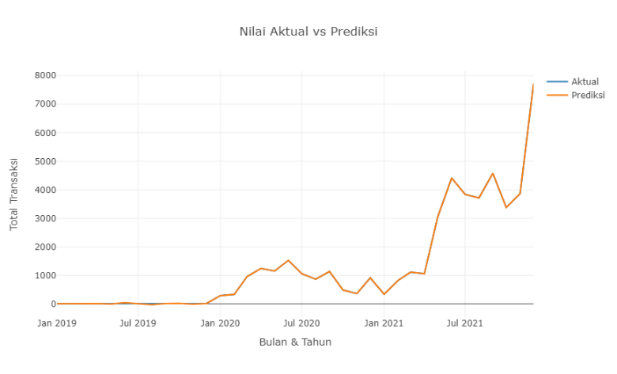
## Pembahasan

Pada proses sebelumnya telah melakukan dua skenario uji model dengan mencoba parameter bawaan dan juga parameter terbaik yang telah dihasilkan oleh *GridSearchCV*. Berdasarkan skenario uji model yang telah dibuat, didapatkan model terbaik yaitu pada skenario uji model dua. Pemodelan pada skenario tersebut walaupun menghasilkan nilai *error* sedikit dibandingkan skenario uji model satu tetapi model dapat dikatakan lebih baik dari skenario uji model dua. Nilai MAE pada data latih dan data uji masing-masing menghasilkan nilai 0.1353 dan 0.1308, maksud nilai ini menunjukkan antara nilai aktual dengan nilai prediksi memiliki selisih yang sangat rendah. Sedangkan evaluasi menggunakan pada data latih dan data uji masing-masing menunjukkan nilai yang sama yaitu 0,99, nilai ini bermakna bahwa atribut *month* (X1) dan *year* (X2) secara simultan atau bersama-sama berpengaruh terhadap atribut total\_trx\_per\_month\_year (Y) sehingga dapat dikatakan pengaruh yang dihasilkan sangat signifikan. Berdasarkan uji model tersebut, selisih nilai aktual dan nilai prediksi tergolong sangat kecil. Interpretasi perbandingan nilai aktual dan nilai prediksi disajikan dalam bentuk tabel dan visualisasi grafik seperti pada Tabel 11 dan Gambar 8.

Tabel 11. Hasil Perbandingan Nilai Aktual dan Prediksi Uji Model 2

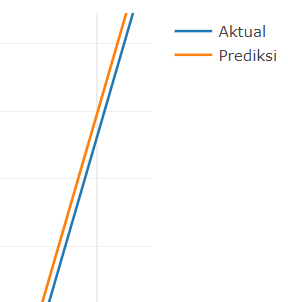
|  |  |  |  |
| --- | --- | --- | --- |
| Tahun | Bulan | Nilai Aktual | Nilai Prediksi |
| 2019 | Januari | 3 | 5.22 |
| Februari | 4 | 7.68 |
| Maret | 10 | 2.93 |
| April | 11 | 14.47 |
| Mei | 7 | -5.75 |
| Juni | 8 | 41.34 |
| Juli | 3 | 3.81 |
| Agustus | 8 | -32.17 |
| September | 15 | 13.13 |
| Oktober | 10 | 16.56 |
| November | 2 | -11.44 |
| Desember | 12 | 12.28 |
| 2020 | Januari | 287 | 286.86 |
| Februari | 328 | 327.84 |
| Maret | 960 | 960.43 |
| April | 1242 | 1241.75 |
| Mei | 1147 | 1147.41 |
| Juni | 1522 | 1521.88 |
| Juli | 1056 | 1055.65 |
| Agustus | 863 | 863.05 |
| September | 1133 | 1133.24 |
| Oktober | 486 | 484.98 |
| November | 356 | 356.85 |
| Desember | 908 | 907.96 |
| 2021 | Januari | 338 | 338.06 |
| Februari | 819 | 819.07 |
| Maret | 1108 | 1107.68 |
| April | 1055 | 1055.21 |
| Mei | 3057 | 3056.89 |
| Juni | 4411 | 4410.97 |
| Juli | 3835 | 3835.12 |
| Agustus | 3713 | 3713.01 |
| September | 4575 | 4574.92 |
| Oktober | 3378 | 3378.15 |
| November | 3864 | 3863.93 |
| Desember | 7725 | 7724.99 |

Pada Tabel 11 nilai prediksi yang dihasilkan dari metode *XGBoost* menunjukkan nilai yang beragam. Nilai prediksi tersebut ada yang lebih tinggi dari nilai aktual, ada yang lebih rendah dari aktual, dan bahkan terdapat nilai yang hampir sama dengan nilai aktual. Hal ini disebabkan karena model regresi pada metode *XGBoost* menghasilkan nilai residu yang berbeda-beda di setiap iterasi nya. Dengan nilai residu atau error yang berbeda tersebut, maka setelah dilakukan perhitungan didapatkan nilai prediksi yang kemungkinan hasilnya akan lebih tinggi, lebih rendah, atau bahkan hampir sama dengan nilai aktual. Berdasarkan nilai tersebut bukan berarti model yang dihasilkan tidak baik, melainkan hal ini dilakukan *XGBoost* untuk menghindari model *overfitting* atau *underfitting*. Dengan pemilihan parameter menggunakan *GridSearchCV*, metode *XGBoost* melakukan regularisasi dan batasan agar mencegah model dari *overfitting* atau *underfitting* sehingga didapatkan model yang baik.



Gambar 8. Plot Visualisasi Perbandingan Nilai Aktuan dan Prediksi Uji Model 2

Pada Gambar 8 dapat dilihat bahwa garis berwarna biru yang mendefinisikan nilai aktual dapat dikatakan hampir menempel dengan nilai prediksi. Tetapi jika dilihat lebih dekat lagi perbedaan nilai tersebut tetap terlihat walaupun selisih jarak nya sangat kecil dan dapat dilihat pada Gambar 9. Hal ini juga ditandai pada Tabel 11bahwa semua nilai prediksi yang dihasilkan terdapat perbedaan dengan nilai aktual.



Gambar 11. Perbedaan Garis Nilai Aktual dan Prediksi Uji Model 2

Berdasarkan hasil tersebut, hal ini dibuktikan dengan beberapa penelitian terdahulu yang menggunakan metode serupa. Pada penelitian yang dilakukan Dairu dan Shilong (2021), nilai RMSSE yang dihasilkan sebesar 0,655. Sedangkan pada penelitian yang dikerjakan oleh Ranjitha dan Spandana (2021), nilai MAE yang dihasilkan sebesar 0,029. Penelitian yang dilakukan Zhang dkk (2021), MAE yang dihasilkan dari dua *dataset* berbeda masing-masing sebesar 0,178 dan 0,046. Metode *XGBoost* terbukti menghasilkan nilai *error* terendah yang didukung dengan beberapa penelitian terdahulu yang telah dijelaskan sebelumnya dan dapat dikatakan juga model dari metode *XGBoost* bisa diimplementasikan untuk prediksi regresi total transaksi.

# penutup

## kesimpulan

Berdasarkan hasil penelitian dan analisis yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut:

1. Berdasarkan pengujian yang dilakukan, pembuatan model menggunakan *XGBoost* memberikan kinerja yang baik pada penelitian ini. Hal ini ditandai dengan hasil MAE dan yang diberikan masing-masing sebesar 0,1308, dan 0,99.
2. Evaluasi performa model terbaik ada pada skenario uji model dua dengan nilai MAE pada data latih dan data uji masing-masing menghasilkan nilai sebesar 0,1353 dan 0,1308. Sedangkan evaluasi menggunakan pada data latih dan data uji masing-masing menunjukkan nilai yang sama yaitu 0,99.

## saran

Berdasarkan hasil yang diperoleh yang dilakukan selama penelitian ini, peneliti memberikan saran kepada peneliti selanjutnya dalam bidang dan objek penelitian yang sama sebagai berikut:

1. Penelitian selanjutnya dapat menambah analisis prediksi transaksi dengan deret waktu dapat dilihat per hari, per minggu, per bulan, dan per tahun sehingga prediksi selanjutnya dapat lebih informatif.
2. Pengujian model yang dilakukan dapat menambah parameter yang lebih bervariasi lagi jika menggunakan metode serupa.
3. Analisis dan visualisasi yang dilakukan masih sangat sederhana, maka perlu dikembangkan kembali. Untuk peneliti selanjutnya dapat mengembangkan keluaran yang dihasilkan dalam bentuk sistem atau *dashboard* agar informasi yang ditampilkan mempermudah pembaca untuk memahaminya.

# referensi

[1] Alpaydin Ethem, “Introduction to Machine Learning,” 2020.

[2] H. S. R. Rajula, G. Verlato, M. Manchia, N. Antonucci, and V. Fanos, “Comparison of conventional statistical methods with machine learning in medicine: Diagnosis, drug development, and treatment,” *Medicina (Lithuania)*, vol. 56, no. 9, pp. 1–10, Sep. 2020, doi: 10.3390/medicina56090455.

[3] D. Kurniawan, *Pengenalan Machine Learning dengan Python*. PT Elex Media Komputindo, 2020.

[4] N. Wayan Wardani, “Penerapan Data Mining Dalam Analytic CRM,” 2021. [Online]. Available: https://www.researchgate.net/publication/351776255

[5] Y. Niu, “Walmart Sales Forecasting using XGBoost algorithm and Feature engineering,” in *Proceedings - 2020 International Conference on Big Data and Artificial Intelligence and Software Engineering, ICBASE 2020*, Oct. 2020, pp. 458–461. doi: 10.1109/ICBASE51474.2020.00103.

[6] X. Dairu and Z. Shilong, “Machine Learning Model for Sales Forecasting by Using XGBoost,” in *2021 IEEE International Conference on Consumer Electronics and Computer Engineering, ICCECE 2021*, Jan. 2021, pp. 480–483. doi: 10.1109/ICCECE51280.2021.9342304.

[7] T. Chen and T. He, “xgboost: eXtreme Gradient Boosting,” 2017.

[8] “Pengadaan Barang dan Jasa Pemerintah di Marketplace PaDi UMKM,” May 12, 2022. https://padiumkm.id/ (accessed May 12, 2022).

[9] “Frequently Asked Question,” May 12, 2022. https://control.padiumkm.id/faq (accessed May 12, 2022).

[10] Daqiqil Ibnu, “Machine Learning: Teori, Studi Kasus, dan Implementasi Menggunakan Python,” 2021.

[11] T. Indarwati, T. Irawati, and E. Rimawati, “PENGGUNAAN METODE LINEAR REGRESSION UNTUK PREDIKSI PENJUALAN SMARTPHONE,” *Jurnal Teknologi Informasi dan Komunikasi (TIKomSiN)*, vol. 6, no. 2, Jan. 2019, doi: 10.30646/tikomsin.v6i2.369.

[12] F. Nargesian, H. Samulowitz, U. Khurana, E. B. Khalil, and D. Turaga, “Learning feature engineering for classification,” in *IJCAI International Joint Conference on Artificial Intelligence*, 2017, vol. 0, pp. 2529–2535. doi: 10.24963/ijcai.2017/352.

[13] C. M. Sitorus, A. Rizal, and M. Jajuli, “Prediksi Risiko Perjalanan Transportasi Online Dari Data Telematik Menggunakan Algoritma Support Vector Machine,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 6, no. 2, Aug. 2020, doi: 10.28932/jutisi.v6i2.2672.

[14] C. M. Sitorus, A. Rizal, and M. Jajuli, “Prediksi Risiko Perjalanan Transportasi Online Dari Data Telematik Menggunakan Algoritma Support Vector Machine,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 6, no. 2, 2020, doi: 10.28932/jutisi.v6i2.2672.

[15] A. A. Firdaus, A. Komarudin, M. P. Statistika, F. Matematika, D. Ilmu, and P. Alam, “Klasifikasi Pemegang Polis Menggunakan Metode XGBoost,” 2021, doi: 10.29313/.v0i0.30320.

[16] Daoud Al Essam, “Comparison between XGBoost, LightGBM and  CatBoost Using a Home Credit Dataset,” 2019.

[17] J. Qi, J. Du, S. M. Siniscalchi, X. Ma, and C. H. Lee, “On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression,” *IEEE Signal Processing Letters*, vol. 27, pp. 1485–1489, 2020, doi: 10.1109/LSP.2020.3016837.

[18] E. Kasuya, “On the use of r and r squared in correlation and regression,” *Ecological Research*, vol. 34, no. 1, pp. 235–236, Jan. 2019, doi: 10.1111/1440-1703.1011.