

ESTIMASI RENCANA ANGGARAN BIAYA PROYEK TELEKOMUNIKASI BERBASIS ALGORITMA CATBOOST DAN DATA BOQ

Galbi Nadifah¹⁾, Desi Anggreani²⁾, Rizki Yusliana Bakti³⁾, Muhammad Faisal⁴⁾,
Lukman Anas⁵⁾

1. Teknik, Universitas Muhammadiyah Makassar
email: 105841116322@student.unismuh.ac.id
2. Teknik, Universitas Muhammadiyah Makassar
email: desianggreani@unismuh.ac.id
3. Teknik, Universitas Muhammadiyah Makassar
email: rizkiyusliana@unismuh.ac.id
4. Teknik, Universitas Muhammadiyah Makassar
email: muhfaisal@unismuh.ac.id
5. Teknik, Universitas Muhammadiyah Makassar
email: lukmananas@unismuh.ac.id

Estimasi Rencana Anggaran Biaya (RAB) yang akurat merupakan elemen krusial dalam keberhasilan proyek infrastruktur telekomunikasi. Penentuan harga satuan pada dokumen Bill of Quantities (BOQ) secara konvensional sering kali tidak efisien dan subjektif. Tantangan utama otomatisasi estimasi ini adalah tingginya kardinalitas fitur kategorikal berupa teks deskriptif. Penelitian ini mengusulkan penerapan algoritma CatBoost untuk memprediksi harga satuan pekerjaan berbasis data BOQ. Tahapan penelitian meliputi pembersihan data historis sebanyak 20.611 item, transformasi logaritma natural pada variabel target, serta pelatihan model dengan pembagian data 80% latih dan 20% uji. Hasil eksperimen menunjukkan CatBoost mampu menghasilkan Coefficient of Determination (R^2) sebesar 97,66% dan Mean Absolute Error (MAE) sebesar Rp 13.289. Kinerja ini unggul dibandingkan algoritma pembanding XGBoost (R^2 85,93% dan MAE Rp 23.614). Validasi manual mengonfirmasi rasio kesalahan prediksi hanya 0,05 dari total nilai proyek, yang membuktikan kelayakan model untuk otomatisasi RAB secara presisi.

Kata Kunci: RAB, Bill of Quantities, CatBoost, Machine Learning, Estimasi Biaya.

A. PENDAHULUAN

Perencanaan biaya yang presisi pada industri infrastruktur telekomunikasi menjadi penentu utama profitabilitas dan keberlanjutan proyek [1]. Penyusunan Rencana Anggaran Biaya (RAB) berbasis *Bill of Quantities* (BOQ) secara konvensional kerap memicu *cost overrun* atau kegagalan penawaran tender akibat

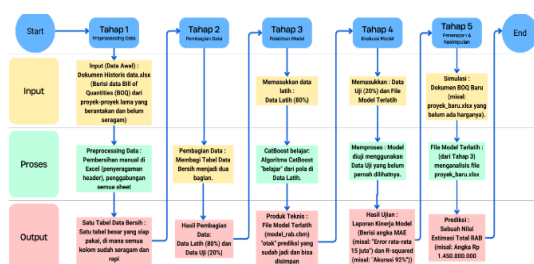
ketergantungan pada kalkulasi manual, pencarian data historis (*lookup*), serta estimasi yang subjektif [1], [2]. Pendekatan *Machine Learning* berbasis data (*data-driven*) seperti *Support Vector Regression* (SVR) dan *Artificial Neural Networks* (ANN) telah banyak diterapkan untuk memprediksi biaya konstruksi [3], [4]. Namun, kendala utama pada dataset BOQ adalah dominasi variabel

kategorikal *non-numerik* ber-kardinalitas tinggi (*high cardinality*), seperti deskripsi teks pekerjaan dan lokasi [5]. Algoritma berbasis pohon keputusan konvensional seperti XGBoost umumnya membutuhkan konversi data melalui label *encoding* atau *one-hot encoding*, yang berisiko memperbesar dimensi data serta memicu *overfitting* [5].

Untuk mengatasi keterbatasan tersebut, penelitian ini menerapkan algoritma CatBoost (*Categorical Boosting*) [6]. CatBoost memanfaatkan teknik *Ordered Target Statistics* yang mampu mengolah fitur kategorikal secara langsung tanpa konversi manual yang merusak struktur data, sehingga terbukti efektif menekan *overfitting* pada dataset *non-numerik* [6], [7]. Penelitian ini bertujuan membangun dan memvalidasi model prediksi harga satuan BOQ otomatis berbasis CatBoost serta membandingkan kinerjanya terhadap algoritma pembandingan XGBoost [5], [6].

B. METODE PENELITIAN

Penelitian ini dilakukan melalui beberapa tahapan sistematis untuk memastikan model yang dihasilkan optimal. Tahapan dimulai dari pengumpulan data, pembersihan data, transformasi fitur, pembagian data latih dan uji, hingga evaluasi model. Alur metodologi penelitian ditunjukkan pada Gambar 1.



Gambar 1. Desain Perancangan Sistem

Data yang digunakan dalam penelitian ini adalah data historis proyek infrastruktur telekomunikasi yang mencakup 20.611 item pekerjaan. Variabel independen (*features*) meliputi atribut kategorikal: *Designator*, Uraian Pekerjaan, Satuan, Jenis, dan Wilayah. Variabel dependen (*target*) adalah Harga Satuan.

Tahapan **preprocessing** mencakup:

1. Pembersihan Data: Menghapus data duplikat dan data *outlier*.
2. Transformasi Logaritma: Karena distribusi harga sangat timpang (*skewed*), dilakukan transformasi logaritma natural:

$$y' = \ln(1 + y) \quad (1)$$

Dimana y adalah harga satuan asli dan y' adalah target pelatihan. Hal ini bertujuan untuk menstabilkan *varians error* [8].

Penelitian ini membandingkan dua algoritma Gradient Boosting:

1. *CatBoost* (Usulan): Menggunakan parameter loss function RMSE, dengan fitur otomatisasi penanganan kategori. Pelatihan dilakukan selama 1.000 iterasi (*epoch*) untuk memantau penurunan loss.
2. *XGBoost* (Baseline): Sebagai pembandingan standar industri.

Model dievaluasi menggunakan metode *Hold-out Validation* (80% latih, 20% uji) dengan metrik utama R^2 (Koefisien Determinasi) dan MAE (*Mean Absolute Error*).

1) Kerangka Kerja Penelitian

Penelitian ini dilakukan dengan pendekatan kuantitatif eksperimental untuk membangun model prediksi harga satuan. Tahapan penelitian disusun secara sistematis mulai dari pengumpulan data,

pra-pemrosesan, pelatihan model, hingga evaluasi. Alur kerja penelitian ini divisualisasikan dalam Gambar 1.

2) Pengumpulan Data

Data yang digunakan merupakan data sekunder yang berasal dari riwayat proyek infrastruktur telekomunikasi (*Historical Project Data*). Dataset terdiri dari 20.611 baris data item pekerjaan.

Setiap entitas data memiliki variabel independen (*features*) yang bersifat kategorikal dan satu variabel dependen (*target*) numerik. Rincian variabel yang digunakan dalam pemodelan adalah:

- Fitur Input (X): *Designator* (Kode Pekerjaan), Uraian Pekerjaan, Satuan, Jenis (Material/Jasa), District, Sub-District, dan Wilayah.
- Fitur Target (Y): Harga Satuan (Rupiah).

3) Pra-pemrosesan Data

Mengingat karakteristik data *BOQ* yang memiliki kardinalitas tinggi (*high-cardinality*) dan variasi harga yang ekstrim, dilakukan tahapan **pra-pemrosesan** sebagai berikut:

- Pembersihan Data: Menghapus data duplikat dan data *noise* (harga bernilai nol atau negatif) untuk menjaga kualitas pembelajaran model.
- Penanganan Kategori: Untuk fitur kategorikal yang memiliki nilai kosong (*missing values*), dilakukan imputasi dengan label "UNKNOWN" agar informasi tetap dapat diproses.
- Transformasi Logaritma: Distribusi harga satuan pada data asli memiliki kemiringan (*skewness*) yang tinggi. Untuk menstabilkan varians dan meningkatkan performa algoritma regresi, variabel target (*y*) ditransformasi menggunakan fungsi logaritma natural.

4) Arsitektur Model

Penelitian ini menerapkan dan membandingkan dua algoritma *Gradient Boosting*:

- CatBoost* (*Proposed Model*): Model ini dipilih sebagai metode utama karena keunggulannya dalam menangani fitur kategorikal secara otomatis menggunakan teknik *Ordered Target Statistics*. Model dilatih dengan parameter *loss function* *RMSE* (*Root Mean Square Error*) dan iterasi sebanyak 1.000 *epoch*.
- XGBoost* (*Baseline Model*): Digunakan sebagai pembandingan standar. Karena *XGBoost* tidak menangani data teks secara native, dilakukan proses Label Encoding pada fitur kategorikal sebelum data dimasukkan ke dalam model.

5) Evaluasi Dan Validasi

Dataset dibagi menggunakan metode *Hold-out Validation* dengan proporsi 80% data latih dan 20% data uji (4.123 data). Kinerja model diukur menggunakan dua pendekatan:

- Evaluasi Statistik: Menggunakan metrik R^2 (*Coefficient of Determination*)

$$R^2 = 1 - \frac{\sum (y_{\text{actual}} - y_{\text{prediksi}})^2}{\sum (y_{\text{actual}} - y_{\text{rata-rata}})^2}$$

untuk melihat kecocokan model, dan *MAE* (*Mean Absolute Error*) untuk menghitung rata-rata kesalahan dalam satuan Rupiah.

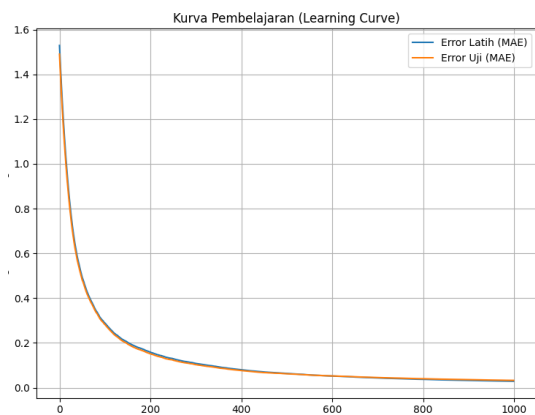
$$MAE = \frac{1}{n} \sum_{i=1}^n |y_{\text{actual}} - y_{\text{prediksi}}|$$

- Validasi Aritmatika Manual: Melakukan perhitungan manual terhadap rasio total kesalahan ($\sum |error|$) dibandingkan dengan total harga proyek ($\sum y$) untuk memastikan validitas model dari perspektif bisnis/anggaran.

C. HASIL DAN PEMBAHASAN

1. Analisis Proses Penelitian Model

Pelatihan model dilakukan menggunakan *dataset* yang telah melalui tahap *pra-pemrosesan* dan transformasi logaritma. Untuk memastikan model dapat mempelajari pola data dengan baik tanpa mengalami *overfitting*, proses pelatihan dipantau menggunakan metrik evaluasi *RMSE (Root Mean Square Error)* pada setiap iterasi (*epoch*).



Gambar 2. Learning Curve CatBoost

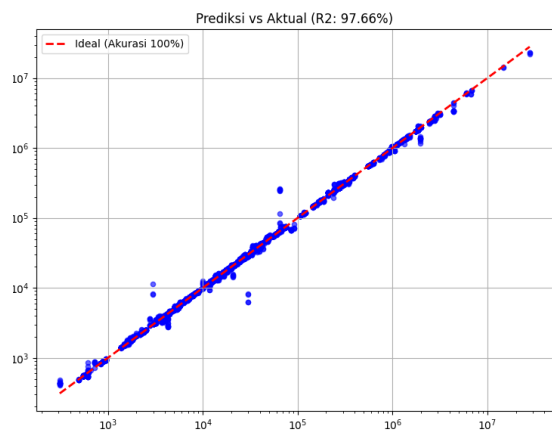
Selain validasi visual melalui grafik, stabilitas proses pelatihan juga dapat diamati secara numerik melalui rekam jejak penurunan tingkat kesalahan (*error*) pada setiap tahapan iterasi. Rincian penurunan nilai *Mean Absolute Error (MAE)* pada data latih dan data uji yang dipantau secara berkala disajikan pada Tabel 1 berikut :

Tabel 1. Iterasi CatBoost

Iterasi	Training MAE	Testing MAE	Selisih
1	1.52901	1.49016	0.03885
200	0.15900	0.15195	0.00705
400	0.08014	0.07651	0.00363
600	0.05187	0.05226	0.00039
800	0.03724	0.03988	0.00264

Iterasi	Training MAE	Testing MAE	Selisih
1000	0.02863	0.03257	0.00393

Selanjutnya, untuk memverifikasi tingkat kesesuaian antara nilai prediksi model dengan nilai aktual di lapangan, dilakukan pemetaan distribusi data menggunakan grafik tebar. Pola sebaran titik prediksi terhadap garis referensi ideal ($y=x$) divisualisasikan pada Gambar 3 berikut :



Gambar 3. Scatter Plot CatBoost

Rincian capaian nilai metrik evaluasi *MAE* dan R^2 dari model *CatBoost* disajikan secara lengkap pada Tabel 2 berikut :

Tabel 2. Hasil Evaluasi Kinerja Model CatBoost

Metrik Evaluasi	Nilai Capaian	Interpretasi
Mean Absolute Error (MAE)	Rp 13.289	Jika dibandingkan dengan rata-rata nilai proyek dalam dataset yang mencapai Rp 235.845, besaran <i>error</i> ini tergolong sangat minor hanya

Metrik Evaluasi	Nilai Capaian	Interpretasi
		0,05 bagian dari skala data.
<i>R-Squared</i> (R^2)	97,66%	Model berhasil menjelaskan 97,66% variasi pola pembentukan harga total RAB.

Untuk memvalidasi kebenaran nilai evaluasi kinerja yang dihasilkan oleh sistem, dilakukan verifikasi perhitungan secara manual (aritmatika) berdasarkan rumus standar statistik. Validasi ini berfokus pada dua indikator kinerja, yakni *Mean Absolute Error* (MAE) yang merepresentasikan rata-rata penyimpangan harga dalam satuan Rupiah, dan *R-Squared* (R^2) yang merepresentasikan ketepatan model dalam mengikuti pola data. Berikut adalah persamaan matematis yang digunakan sebagai acuan perhitungan manual tersebut.

Diketahui :

1. Jumlah Data (n): 4123
2. Total Harga Aktual (Σy): 976.513.130,00
3. Total Kesalahan Mutlak ($\Sigma |e|$): 54.792.541,43
4. Total SS_{res} : 100.516.908.939.252,59
5. Total SS_{tot} : 4.295.540.809.369.178,50

MAE dihitung dengan membagi total selisih mutlak dengan jumlah data.

$$MAE = \frac{\sum |y_{prediksi} - y_{aktual}|}{n}$$

$$MAE = \frac{\text{Total Kesalahan Mutlak}}{\text{Jumlah Data}}$$

$$MAE = \frac{54.792.541,43}{4123}$$

$$MAE = 13.289,48$$

Hasil perhitungan manual menunjukkan nilai MAE sebesar Rp 13.289,48 yang sesuai dengan hasil keluaran sistem.

Perhitungan Rasio Kesalahan (*Error Ratio*) dihitung untuk melihat persentase penyimpangan prediksi terhadap total nilai proyek sebenarnya.

$$Rasio = \frac{\sum |e|}{\sum y}$$

$$Rasio = \frac{\text{Total Kesalahan Mutlak}}{\text{Total Harga Aktual}}$$

$$Rasio = \frac{54.792.541,43}{976.513.130,00}$$

$$Rasio = 0,056110 \text{ atau } 0,05$$

Hasil ini menunjukkan bahwa rata-rata kesalahan prediksi model hanya sebesar 0,05 dari total nilai harga asli.

Perhitungan Koefisien Determinasi (R^2) Nilai R^2 dihitung untuk mengukur seberapa baik model dalam menjelaskan variasi data.

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

$$R^2 = 1 - \left(\frac{100.516.908.939.252,59}{4.295.540.809.396.178,50} \right)$$

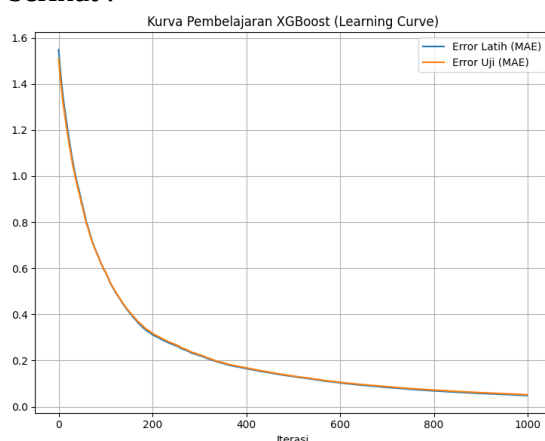
$$R^2 = 1 - 0,02339$$

$$R^2 = 0,9766$$

Hasil perhitungan manual menghasilkan nilai R^2 sebesar 0,9766 (97,66%), yang mengonfirmasi bahwa model memiliki tingkat akurasi yang sangat tinggi.

Sebagai validasi keunggulan metode, dilakukan perbandingan kinerja dengan algoritma *XGBoost* (*eXtreme Gradient Boosting*). Perbedaan mendasar dalam eksperimen ini terletak pada penanganan fitur kategorikal, *CatBoost* menggunakan *Ordered Target Statistics* (otomatis), sedangkan *XGBoost* menggunakan *Label Encoding* (manual).

Dampak dari penggunaan metode *Label Encoding* tersebut terhadap stabilitas proses pelatihan dapat diamati melalui visualisasi *Learning Curve*. Dinamika pergerakan *error* (MAE) pada model pembandingan *XGBoost* selama proses iterasi ditampilkan pada Gambar 4 berikut :



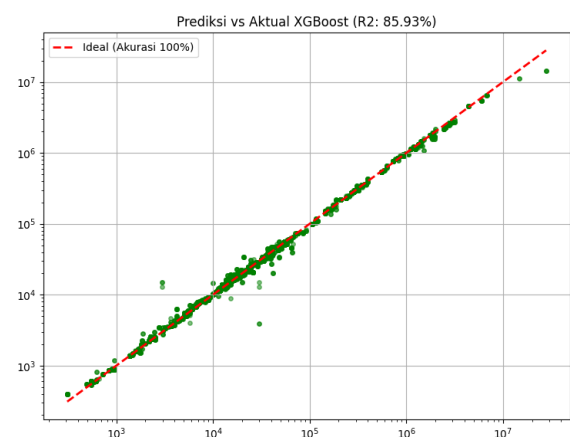
Gambar 4. Learning Curve XGBoost

Selain analisis visual, rincian numerik mengenai penurunan tingkat kesalahan juga direkam untuk membandingkan kecepatan konvergensi antara kedua model. Data riwayat penurunan *Mean Absolute Error* (MAE) *XGBoost* pada interval iterasi tertentu dirangkum dalam Tabel 3 di bawah ini:

Tabel 3. Iterasi XGBoost

Iterasi	Training MAE	Testing MAE	Selisih
1	1.54855	1.51020	0.03835
200	0.31367	0.31943	0.00576
400	0.16575	0.16875	0.00300
600	0.10352	0.10612	0.00260
800	0.06908	0.07268	0.00360
1000	0.04849	0.05180	0.00331

Selanjutnya, untuk memvalidasi tingkat akurasi akhir, dilakukan pemetaan distribusi nilai prediksi *XGBoost* terhadap nilai aktual di lapangan. Pola sebaran data yang dihasilkan oleh model pembandingan ini diperlihatkan melalui grafik *Scatter Plot* pada Gambar 5:



Gambar 5. Scatter Plot XGBoost

Guna memvalidasi presisi dan reliabilitas model prediksi, penelitian ini mengadopsi dua indikator evaluasi statistik utama, yakni *Mean Absolute*

Error (MAE) dan Koefisien Determinasi (R^2). MAE berfungsi untuk mengkuantifikasi rata-rata penyimpangan absolut antara hasil prediksi terhadap data aktual, sementara R^2 merepresentasikan proporsi variabilitas data yang mampu dijelaskan oleh model. Berikut adalah persamaan matematis dari kedua parameter tersebut.

Diketahui :

1. Jumlah Data (n): 4123
2. Total Harga Aktual (Σy): 976.513.130,00
3. Rata-rata Harga Aktual: 236.845,29
4. Total Kesalahan Mutlak ($\Sigma |e|$): 97.361.319,15
5. Total SS_{res} : 604.273.706.693.570,00
6. Total SS_{tot} : 4.295.540.809.369.178,50

Perhitungan Mean Absolute Error (MAE) dihitung dengan membagi total selisih mutlak dengan jumlah data.

$$MAE = \frac{\Sigma |y_{prediksi} - y_{aktual}|}{n}$$

$$MAE = \frac{\text{Total Kesalahan Mutlak}}{\text{Jumlah Data}}$$

$$MAE = \frac{97.361.319,15}{4123}$$

$$MAE = 23.614,19$$

Hasil perhitungan manual menunjukkan nilai MAE sebesar Rp 23.614,19 yang sesuai dengan hasil keluaran sistem.

Rasio kesalahan dihitung untuk melihat persentase penyimpangan prediksi terhadap total nilai proyek sebenarnya.

$$Rasio = \frac{\Sigma |e|}{\Sigma y}$$

$$Rasio = \frac{\text{Total Kesalahan Mutlak}}{\text{Total Harga Aktual}}$$

$$Rasio = \frac{97.361.319,15}{976.513.130,00}$$

$$Rasio = 0,099703 \text{ atau } 0,09$$

Hasil ini menunjukkan bahwa rata-rata kesalahan prediksi model hanya sebesar 0,09 dari total nilai harga asli.

Perhitungan Koefisien Determinasi (R^2) Nilai R^2 dihitung untuk mengukur seberapa baik model dalam menjelaskan variasi data.

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

$$R^2 = 1 - \left(\frac{604.273.706.693.570,00}{4.295.540.809.369.178,50} \right)$$

$$R^2 = 1 - 0,140675$$

$$R^2 = 0,8593$$

Hasil perhitungan manual menghasilkan nilai R^2 sebesar 0,8593 (85,93%).

2. Perbandingan Kinerja Model

Untuk mengukur efektivitas model yang diusulkan, kinerja CatBoost dibandingkan dengan algoritma *baseline* XGBoost menggunakan data uji yang

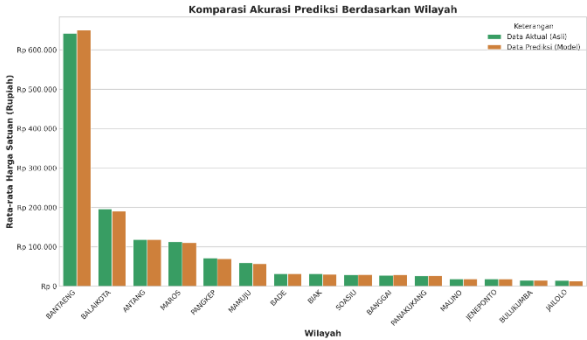
sama (4.123 sampel). Hasil evaluasi kuantitatif berdasarkan metrik statistik disajikan pada Tabel 4 :

Tabel 4. Perbandingan Kinerja Algoritma		
Parameter	CatBoost (Usulan)	XGBoost (Pembandingan)
Metode Kategori	<i>Ordered Target Statistics</i>	<i>Label Encoding</i>
Akurasi (R^2)	97,66%	85.93%
Error (MAE)	Rp 13.289	Rp 23.614

CatBoost unggul dengan nilai R^2 mencapai 97,66%, lebih tinggi dibandingkan XGBoost 0,84%. Selain itu, *CatBoost* mampu menekan rata-rata kesalahan prediksi (MAE) menjadi Rp 13.289, yang lebih rendah dibandingkan XGBoost Rp 23.614. Hal ini membuktikan bahwa penanganan fitur kategorikal menggunakan metode *Ordered Target Statistics* pada CatBoost lebih efektif dibandingkan metode konvensional.

3. Analisis Konsistensi Wilayah

Selain evaluasi global, dilakukan pula analisis mendalam untuk melihat konsistensi model di berbagai wilayah geografis proyek. Hal ini penting untuk memastikan model tidak bias terhadap lokasi tertentu. Gambar 6 menampilkan komparasi visual antara rata-rata harga aktual dan prediksi CatBoost per wilayah.



Gambar 6. Grafik Perbandingan Rata-rata Harga Satuan Aktual dan Prediksi per Wilayah

Berdasarkan Gambar 6, dapat diamati bahwa model menunjukkan kinerja yang sangat konsisten di seluruh wilayah. Hal ini dibuktikan dengan tinggi batang oranye (Data Prediksi) yang selalu mengikuti pola tinggi batang hijau (Data Aktual) secara presisi. Adapun rincian mengenai selisih antara nilai prediksi dan nilai aktual disajikan pada Tabel 5 berikut:

WIL AYA H	HAR GA_ ASLI	HARG A_PRE DIKSI	SELI SIH (Rp)	ERRO R
BAN TAE NG	Rp 642,0 89	Rp 650,49 0	Rp 8,401	0,0131
BAL AIKO TA	Rp 195,8 36	Rp 189,74 9	Rp 6,087	- 0,0311
ANT ANG	Rp 118,0 70	Rp 117,83 6	Rp 234	- 0,0020
MAR OS	Rp 111,9 59	Rp 110,29 0	Rp 1,669	- 0,0149
PAN GKEP	Rp 71,67 7	Rp 69,727	Rp 1,950	- 0,0272

WIL AYA H	HAR GA_ ASLI	HARG A_PRE DIKSI	SELI SIH (Rp)	ERRO R
MAM UJU	Rp 58,84 5	Rp 56,618	Rp - 2,227	0,0378
BAD E	Rp 31,19 0	Rp 31,336	Rp 146	0,0047
BIAK	Rp 31,19 0	Rp 30,207	Rp - 983	0,0315
SOAS IU	Rp 28,96 0	Rp 29,757	Rp 797	0,0275
BAN GGAI	Rp 27,47 0	Rp 28,760	Rp 1,290	0,0470
PAN AKU KAN G	Rp 26,36 1	Rp 26,608	Rp 247	0,0094
MALI NO	Rp 18,86 7	Rp 18,821	Rp - 46	0,0025
JENE PONT O	Rp 18,86 7	Rp 18,483	Rp - 384	0,0204
BUL UKU MBA	Rp 14,57 8	Rp 14,476	Rp - 102	0,0070
JAIL OLO	Rp 14,02 6	Rp 13,723	Rp - 303	0,0216

Data pada Tabel 5 menunjukkan bahwa rasio kesalahan (*error ratio*) di hampir seluruh wilayah berada di bawah angka 0,05 (5%). Bahkan di wilayah "ANTANG", rasionya sangat kecil yaitu 0,0020 (0,2%). Konsistensi ini menegaskan bahwa model dapat

diandalkan untuk implementasi di berbagai distrik tanpa perlu penyesuaian ulang yang signifikan.

D. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian dan evaluasi yang telah dilakukan, penerapan algoritma *CatBoost* dengan fitur *Ordered Target Statistics* terbukti sangat efektif untuk otomatisasi estimasi RAB pada proyek telekomunikasi. Strategi pra-pemrosesan yang ketat berhasil menjaga integritas 20.611 data historis, menghasilkan model dengan kinerja superior dibandingkan algoritma *baseline XGBoost*. Hal ini dibuktikan dengan capaian nilai koefisien determinasi (R^2) sebesar 97,66% dan tingkat kesalahan rata-rata (MAE) yang sangat rendah yaitu Rp 13.289 (rasio *error* 0,05). Hasil validasi ini mengonfirmasi bahwa *CatBoost* mampu menangani kompleksitas data *Bill of Quantities* (BOQ) yang memiliki kardinalitas tinggi secara lebih presisi dan efisien dibandingkan metode konvensional.

Meskipun model telah menunjukkan kinerja yang memuaskan, pengembangan lebih lanjut masih diperlukan untuk memaksimalkan kegunaan praktisnya. Penelitian mendatang disarankan untuk mentransformasi model ini ke dalam bentuk antarmuka pengguna berbasis web atau desktop (GUI) agar lebih mudah dioperasikan oleh staf *non-teknis* di lapangan. Selain itu, akurasi prediksi dapat ditingkatkan lebih jauh dengan mengintegrasikan variabel eksternal yang dinamis, seperti jarak logistik, kondisi topografi lokasi proyek, serta fluktuasi inflasi harga material tahunan, guna meningkatkan sensitivitas sistem terhadap faktor geografis dan ekonomi makro.

E. REFERENSI

- [1] I. A. R. Widianty, "Analisis Rencana Anggaran Biaya (RAB) Pada Proyek Pembangunan Gedung," *Jurnal Konstruksi*, vol. 14, no. 1, 2016.
- [2] S. Tayefeh Hashemi, O. M. Ebadati, and H. Kaur, "Cost estimation and prediction in construction projects: a systematic review on machine learning techniques," *SN Applied Sciences*, vol. 2, no. 10, 2020.
- [3] A. E. Elfaki, S. Alatawi, and E. Abushandi, "Using Intelligent Techniques in Construction Project Cost Estimation: 10-Year Review," *Advances in Civil Engineering*, vol. 2014, 2014.
- [4] F. Antoniou and K. Konstantinidis, "Client-Oriented Highway Construction Cost Estimation Models Using Machine Learning," *Applied Sciences*, vol. 15, no. 18, 2025.
- [5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.
- [6] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: unbiased boosting with categorical features," *Advances in Neural Information Processing Systems*, vol. 31, 2018.
- [7] J. T. Hancock and T. M. Khoshgoftaar, "CatBoost for big data: an interdisciplinary review," *Journal of Big Data*, vol. 7, no. 1, 2020.
- [8] B. So, "Enhanced gradient boosting for zero-inflated insurance claims and comparative analysis of CatBoost, XGBoost, and LightGBM," *Scandinavian Actuarial Journal*, no. 10, pp. 1013–1035, 2024.
- [9] F. Yusfida, "A Hybrid Approach for Recommender Systems Based on Alternating Least Squares and CatBoost," *Jurnal Teknik Informatika (Jutif)*, vol. 6, no. 4, 2025.
- [10] S. M. Robu, "A Review on Artificial Intelligence Technologies in Cost Estimation of Construction Projects," *Discover Civil Engineering*, vol. 1, no. 16, 2024.
- [11] K. K. K. Al-Bayati and A. M. J. Al-Khafaji, "Review of Artificial Intelligence Techniques for Cost Estimation in Construction Projects," *International Journal of Construction Management*, 2023.
- [12] Y. Wang, "Price Prediction of Second-Hand Housing in Chengdu Based on CatBoost Algorithm," in *2022 4th International Conference on Artificial Intelligence and Advanced Manufacture*, 2022.
- [13] J. Hancock and T. M. Khoshgoftaar, "Performance of CatBoost and XGBoost in Medicare Fraud Detection," in *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2020.
- [14] A. R. Putri, N. A. Santoso, and B. A. Santoso, "Implementasi Algoritma Regresi Linier dan ARIMA untuk Prediksi Harga Emas," *Jurnal Sistem Komputer dan Kecerdasan Buatan*, vol. 6, no. 2, 2023.
- [15] V. K. Gupta and A. Kumar, "Comparative Analysis of Machine Learning Models for Sales Prediction," *International Journal of Computer Applications*, vol. 175, no. 2, 2020.