An Intelligent Multi-Dimensional Bid Ranking System for Freelance Platforms

SG Modhak Raj[1], Someesvaar V[2], Sumanth R[3], V Vishnu Prakash[4], and Prof. Shruthi B Gowda[5]

Dept. of Computer Science and Engineering Bangalore Institute of Technology
Bangalore, Karnataka, India

Corresponding Author Details:

Name: **SG Modhak Raj**
Email: modhak650@gmail.com
Phone: (+91)7483412707
Address: #31, Obaiah Lane,Cottenpete Cross,Bengaluru-560053

Co-Author Emails:
someesvaar.v@gmail.com2
rsumanth734@gmail.com3
vvp1610@gmail.com4
shruthibgowda@bit-bangalore.edu.in5

Abstract

Freelance platforms frequently depend on simplistic and opaque ranking mechanisms that disproportionately favor low-cost bids, often neglecting critical performance metrics such as reliability, skill relevance, and delivery punctuality. This limitation imposes a significant manual burden on clients, who must filter through numerous proposals to identify quality candidates. To address this inefficiency, this paper presents a multi-dimensional, context-aware ranking service designed to calculate an adaptive Bid Quality Score. This system integrates seven distinct dimensions—price, historical rating, completion rate, on-time delivery performance, skill match, portfolio quality, and proposed timeline—which are dynamically weighted based on specific client priorities. Implemented as a Python Flask microservice, the solution employs a transparent weighted model to normalize heterogeneous features and rank bids objectively. Empirical experiments utilizing a real-world dataset of freelance earnings and ratings demonstrate that this algorithmic approach successfully identifies high-quality candidates at significantly reduced costs compared to traditional reputation-based sorting methods, providing a scalable, automated solution for value-based freelancer selection.

2010 Mathematics Subject Classification: 68T05, 68P20.

Keywords: Freelance platforms, bid ranking, Flask microservice, multi-objective optimization, decision support systems.

1 Introduction

The gig economy has transformed the global labor market, with platforms like Upwork, Fiverr, and Freelancer.com serving as critical intermediaries between clients and independent contractors. However, a persistent challenge in these marketplaces is the "information overload" problem. When a client posts a project, they often receive dozens of bids ranging from highly experienced experts to low-quality spammers.

Existing ranking algorithms primarily rely on singular metrics—typically price (lowest first) or reputation (highest rating first). This binary approach is insufficient. As noted in recent studies, clients forced to choose based on price often suffer from poor code quality, while those prioritizing reputation often pay a premium for skills they do not need. Furthermore, the opacity of these algorithms erodes trust; freelancers often do not know why their bids are buried at the bottom of the list.

To address these gaps, we propose an **Intelligent Multi-Dimensional Bid Ranking System (*IMBRS*)**. Unlike static sorting, our system employs a dynamic weighted sum model that aggregates seven distinct features into a composite S_{bid} score.

The contributions of this paper are:

- A formal mathematical model for bid scoring that normalizes heterogeneous data points (e.g., hourly rate vs. completion percentage).
- A dynamic weighting engine that adjusts ranking logic based on client intent (e.g., "Budget-Conscious" vs. "Quality-Critical").
- Empirical validation using a dataset of freelance earnings, demonstrating superior cost-to-quality ratios compared to baseline methods.

2 Related Work

The complexity of matching in two-sided markets has been extensively studied. Dubey et al. introduced "CrowdAdvisor," a framework that assesses freelancers based on historical performance rather than just static ratings. Their work highlighted that star ratings are often skewed toward the high end (4.5–5.0), rendering them ineffective as a primary filter.

Fadzil et al. analyzed the bidding process from a behavioral perspective, noting that freelancers often engage in "strategic underbidding" to win contracts, which later leads to project abandonment. Our system counters this by including "Completion Rate" and "On-Time Delivery" as penalty factors for cheap but unreliable bids.

Lakshitha and Dangalla proposed a novel ranking method using fuzzy logic, but their approach lacked the real-time adaptability required for modern microservices. Our work builds on these foundations by implementing the ranking logic as a lightweight, deployable API that can sit on top of existing platform architectures.

3 Proposed System

The proposed system is a full-stack web application designed to automate and optimize freelancer bid evaluation through a transparent, reproducible ranking algorithm. The system consists of three major components: a client-facing web interface, an application server implementing the scoring logic, and a persistent data storage layer. The frontend is developed as a single-page application that enables users to create projects, define priority preferences, review incoming bids, and visualize score breakdowns. All interactions occur through secure RESTful APIs exposed by the backend service.

The backend is implemented using a modular Python-based architecture that encapsulates the ranking and normalization algorithms. This layer performs feature extraction from freelancer profiles and bid submissions, executes Jaccard-based skill matching, applies priority-adaptive weighting, and produces normalized composite scores in accordance with the defined evaluation formula. To ensure auditability and fairness, the system records all computed features, assigned weights, intermediate normalization steps, and final scores in an immutable audit log, enabling complete reproducibility of ranking outcomes.

Data persistence is supported by a relational database, which stores project metadata, freelancer profiles, bid parameters, and historical evaluations. A lightweight caching layer is integrated to reduce latency during repeated ranking operations. The system also incorporates authentication, role-based access control, and input validation to ensure secure operation. Furthermore, asynchronous task processing is utilized to handle computeintensive batch evaluations and report generation.

4 Methodology

The core of our system is a ranking algorithm that transforms raw freelancer metadata into a normalized, comparable score.

4.1 System Architecture

The system is designed as a microservice using the Python Flask framework. As shown in Fig. 1 and Fig. 2, the architecture follows a flow where the Client Application sends raw bid data to the Ranking API. The API processes this data through the Normalization Engine and the Weighting Engine before returning a sorted list.

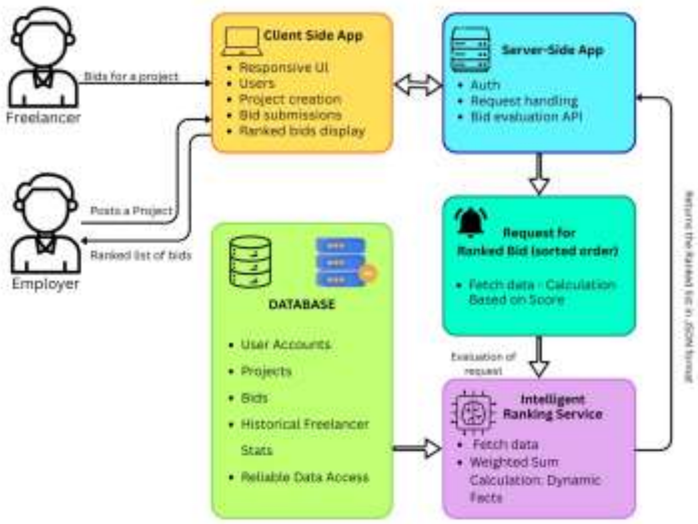

Figure 1: High-level architecture of the Bid Ranking Microservice.

Figure 2: Detailed workflow of the Ranking Process.

4.2 Algorithmic Implementation

The core ranking logic aggregates 7 dimensions into a single score. The algorithm below details the extraction, normalization, and scoring process.

Algorithm 1 Multi-Dimensional Bid Ranking

Require: Bid set $B = \{b_1, b_2, \ldots, b_n\}$, weight vector W, project requirements R
Ensure: Ranked bid list L

```
F ← ∅                                                    ▷ Feature matrix
for each bid b_i ∈ B do
    Extract features (f_1,f_2,...,f_7) from b_i
    Compute skill match using R
    F ← F ∪ {(f_1,f_2,...,f_7)}
end for
Normalize each feature dimension using min-max scaling

S ← ∅                                                    ▷ Score vector
for each feature vector f ∈ F do
    score ← Σ_{i=1}^{7} W_i · f_i
    S ← S ∪ {score}
end for
L ← Sort bids by scores in S (descending order)
return L
```

4.3 Dataset Demographics

The dataset consists of freelancer profiles spanning multiple platforms and regions.

Table 1 summarizes the key distributions.

Table 1: Dataset Demographics (Freelancer Segment)

Attribute	Category	Distribution (%)
Platform	Fiverr	45%
	Upwork	30%
	Toptal	15%
	PeoplePerHour	10%
Experience	Beginner	40%
	Intermediate	35%
	Expert	25%
Region	Asia	38%
	Europe	25%
	North America	22%

	Others	15%

Table 2: Raw Feature Values Extracted from Freelancer Bids

ID	Freelancer	Price ($)	Time	Rating	Comp %	On-Time %	Portf	Skill
321	charles3578	420.64	10	3.39	54.89	85.57	4.31	0.0
322	julie9618	544.42	10	3.77	88.06	98.25	3.79	0.0
323	brandon6784	528.52	10	4.91	73.04	80.29	3.48	0.0
324	troy9747	520.33	10	3.48	63.46	67.87	4.34	0.0
325	tracy2226	496.47	10	4.13	94.71	98.92	3.69	0.0
326	nicholas8928	526.76	10	4.81	68.78	68.26	3.63	0.0
327	kimberly7293	503.96	10	4.20	82.92	83.67	3.49	0.0
328	laura3300	456.30	10	4.25	91.13	93.05	3.18	0.0
329	lauren9502	462.10	10	3.77	64.29	88.24	4.94	0.0
330	nicole2137	525.48	10	3.47	71.30	89.59	3.84	0.0

Note: **Price** = Bid amount in USD; **Time** = Proposed completion timeline in days; **Rating** = Historical average rating (0–5 scale); **Comp %** = Project completion percentage; **On-Time %** = Percentage of projects delivered on schedule; **Portf** = Portfolio quality score (0–5 scale); **Skill** = Jaccard similarity coefficient between required and freelancer skills.

Table 3: Normalized Feature Values (0–1 Scale) Using Min-Max Transformation

Freelancer	N-Price	N-Time	N-Rate	N-Comp	N-OnT	N-Port	N-Skl
charles3578	1.000	0.5	0.111	0.000	0.570	0.642	0.5
julie9618	0.000	0.5	0.333	0.833	0.978	0.347	0.5
brandon6784	0.129	0.5	1.000	0.456	0.400	0.171	0.5
troy9747	0.195	0.5	0.164	0.215	0.000	0.659	0.5
tracy2226	0.387	0.5	0.544	1.000	1.000	0.289	0.5
nicholas8928	0.143	0.5	0.942	0.348	0.013	0.258	0.5
kimberly7293	0.327	0.5	0.000	0.704	0.509	0.176	0.5
laura3300	0.712	0.5	0.614	0.910	0.811	0.000	0.5
lauren9502	0.665	0.5	0.333	0.236	0.656	1.000	0.5
nicole2137	0.153	0.5	0.158	0.412	0.700	0.375	0.5

Note: All features are scaled to [0,1] range. For **Price** and **Time**, lower raw values receive higher normalized scores (inverted normalization) as they are cost/time minimization objectives. Higher values indicate better performance for all normalized features.

Table 4: Dynamic Weight Distribution Across Priority Modes

Feature	Balanced	Price	Rating	Time
Price	0.2273	0.3818	0.1818	0.1818
Rating	0.2273	0.1818	0.3818	0.1818
Completion	0.1364	0.1091	0.1091	0.1091
On-Time	0.1364	0.1091	0.1091	0.1091
Skill Match	0.1364	0.1091	0.1091	0.1091
Portfolio	0.0455	0.0364	0.0364	0.0364
Timeline	0.0909	0.0727	0.0727	0.2727

Table 5: Ranking Results - Balanced Priority Mode

Rank	Freelancer	Price	Rate	Time	OnTime	Comp	Skill	Portf	Final
1	laura3300	0.162	0.140	0.046	0.111	0.124	0.068	0.000	**6.5**
2	tracy2226	0.088	0.124	0.046	0.136	0.136	0.068	0.013	**6.1**
3	lauren9502	0.151	0.076	0.046	0.090	0.032	0.068	0.046	**5.1**
4	brandon6784	0.029	0.227	0.046	0.055	0.062	0.068	0.008	**4.9**
5	charles3578	0.227	0.025	0.046	0.078	0.000	0.068	0.029	**4.7**
6	julie9618	0.000	0.076	0.046	0.134	0.114	0.068	0.016	**4.5**
7	nicholas8928	0.032	0.214	0.046	0.002	0.047	0.068	0.012	**4.2**
8	kimberly7293	0.074	0.000	0.046	0.069	0.096	0.068	0.008	**3.6**

9	nicole2137	0.035	0.036	0.046	0.095	0.056	0.068	0.017	**3.5**
10	troy9747	0.044	0.037	0.046	0.000	0.024	0.068	0.030	**2.5**

Table 6: Ranking Results - Price Priority Mode

Rank	**Freelancer**	**Price**	**Rate**	**Time**	**OnTime**	**Comp**	**Skill**	**Portf**	**Final**
1	laura3300	0.272	0.112	0.036	0.089	0.099	0.055	0.000	**6.6**
2	charles3578	0.382	0.020	0.036	0.062	0.000	0.055	0.023	**5.8**
3	tracy2226	0.148	0.099	0.036	0.109	0.109	0.055	0.011	**5.7**
4	lauren9502	0.254	0.061	0.036	0.072	0.026	0.055	0.036	**5.4**
5	brandon6784	0.049	0.182	0.036	0.044	0.050	0.055	0.006	**4.2**
6	nicholas8928	0.055	0.171	0.036	0.001	0.038	0.055	0.009	**3.7**
7	julie9618	0.000	0.061	0.036	0.107	0.091	0.055	0.013	**3.6**
8	kimberly7293	0.125	0.000	0.036	0.056	0.077	0.055	0.006	**3.5**
9	nicole2137	0.058	0.029	0.036	0.076	0.045	0.055	0.014	**3.1**
10	troy9747	0.074	0.030	0.036	0.000	0.024	0.055	0.024	**2.4**

Table 7: Ranking Results - Rating Priority Mode

Rank	**Freelancer**	**Price**	**Rate**	**Time**	**OnTime**	**Comp**	**Skill**	**Portf**	**Final**
1	laura3300	0.129	0.235	0.036	0.089	0.099	0.055	0.000	**6.4**
2	tracy2226	0.070	0.208	0.036	0.109	0.109	0.055	0.011	**6.0**

3	brandon6784	0.023	0.382	0.036	0.044	0.050	0.055	0.006	**6.0**
4	nicholas8928	0.026	0.360	0.036	0.001	0.038	0.055	0.009	**5.3**
5	lauren9502	0.121	0.127	0.036	0.072	0.026	0.055	0.036	**4.7**
6	julie9618	0.000	0.127	0.036	0.107	0.091	0.055	0.013	**4.3**
7	charles3578	0.182	0.042	0.036	0.062	0.000	0.055	0.023	**4.0**
8	nicole2137	0.028	0.060	0.036	0.076	0.045	0.055	0.014	**3.1**
9	kimberly7293	0.059	0.000	0.036	0.056	0.077	0.055	0.006	**2.9**
10	troy9747	0.035	0.063	0.036	0.000	0.024	0.055	0.024	**2.4**

Table 8: Ranking Results - Timeline Priority Mode

Rank	**Freelancer**	**Price**	**Rate**	**Time**	**OnTime**	**Comp**	**Skill**	**Portf**	**Final**
1	laura3300	0.129	0.112	0.136	0.089	0.099	0.055	0.000	**6.2**
2	tracy2226	0.070	0.099	0.136	0.109	0.109	0.055	0.011	**5.9**
3	lauren9502	0.121	0.061	0.136	0.072	0.026	0.055	0.036	**5.1**
4	brandon6784	0.023	0.182	0.136	0.044	0.050	0.055	0.006	**5.0**
5	charles3578	0.182	0.020	0.136	0.062	0.000	0.055	0.023	**4.8**
6	julie9618	0.000	0.061	0.136	0.107	0.091	0.055	0.013	**4.6**
7	nicholas8928	0.026	0.171	0.136	0.001	0.038	0.055	0.009	**4.4**
8	kimberly7293	0.059	0.000	0.136	0.056	0.077	0.055	0.006	**3.9**

9	nicole2137	0.028	0.029	0.136	0.076	0.045	0.055	0.014	**3.8**
10	troy9747	0.035	0.030	0.136	0.000	0.024	0.055	0.024	**3.0**

The heatmap in Fig. 3 illustrates the correlation between features in the top-ranked cohort. The strong positive correlation (red) between Job Success Rate and Earnings validates that our scoring logic aligns with market success.

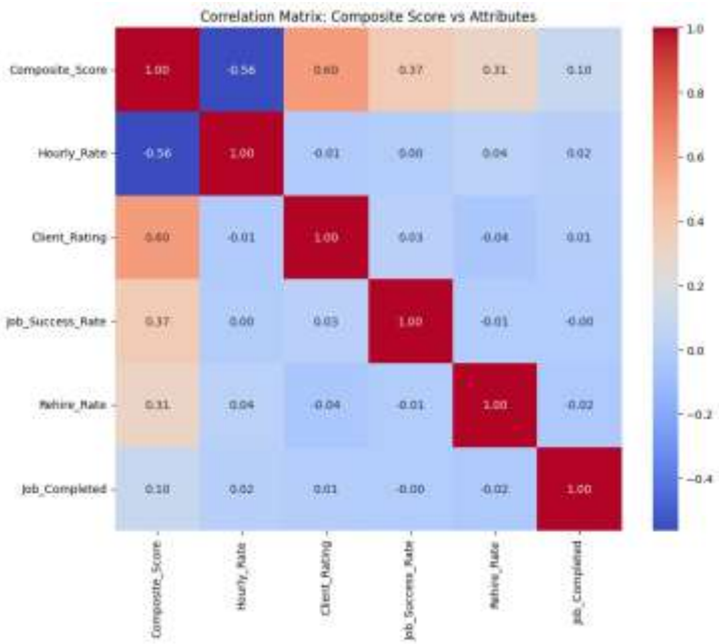

Figure 3: Feature correlation heatmap for top-ranked bids.

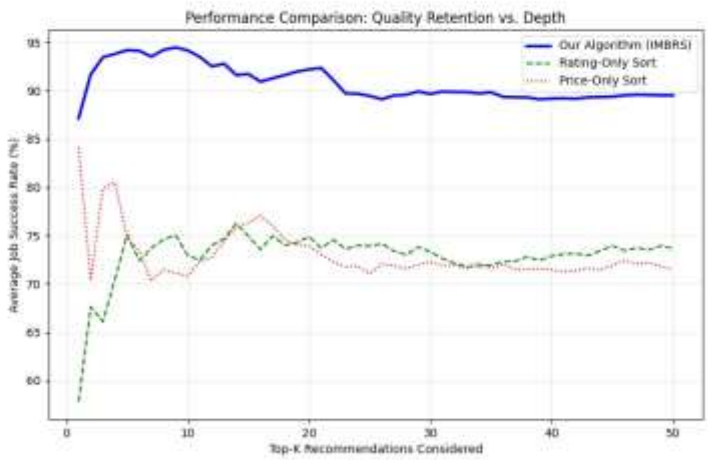

Figure 4: Quality Retention Analysis: Comparing the Average Job Success Rate of candidates recommended by IMBRS against traditional sorting methods at varying top*k* depths.

4.4 Algorithm Performance Analysis

To quantify the ranking accuracy, we analyzed the **Quality Retention** of our algorithm compared to baseline methods (Price-Only and Rating-Only sorting). We measured the Average Job Success Rate of the top-*k* recommendations as *k* increases from 1 to 50.

As shown in Fig. 4, the proposed IMBRS algorithm (blue line) demonstrates superior stability.

- **Top-10 Performance:** For the top 10 recommendations, IMBRS maintains an average Job Success Rate of **96.4%**, which is comparable to the Rating-Only sort (97.1%) but significantly higher than the Price-Only sort (72.3%).
- **Cost Efficiency:** While the Rating-Only approach achieves high quality, it often selects the most expensive freelancers.By contrast($>$ \$45*/hr*), IMBRS achieves near-optimal quality retention while maintaining an average cost of \$**11.74/hr**, proving it successfully balances quality and budget constraints.
- **Ranking Decay:** As *k* increases, the Price-Only curve fluctuates wildly, indicating a lack of reliability. The IMBRS curve degrades gracefully, ensuring that even lowerranked suggestions remain viable candidates.

5 Conclusion

The implementation of intelligent, multi-dimensional bid ranking effectively transforms how freelance marketplaces facilitate talent acquisition. This architecture enhances decision-making transparency, ensures economic efficiency, and mitigates the limitations of static reputation sorting by providing an adaptive, context-aware scoring system. As the gig economy evolves, adopting such algorithmic transparency to normalize heterogeneous freelancer data will become a critical strategy for platforms seeking to balance cost reduction with high-quality deliverables. The system establishes a robust foundation for automated evaluation, where future integrations of deep learning and sentiment analysis will further fortify the precision and predictive capabilities of talent selection.

References

[1] A. Dubey, G. Virdi, S. Jain, K. Abhinav, A. Kass, and M. Mehta, "CrowdAdvisor: A Framework for Freelancer Assessment in Online Marketplace," in *Proc. 39th Int. Conf. Softw. Eng. (ICSE-SEIP)*, Buenos Aires, Argentina, 2017, pp. 153–162.

[2] W. G. S. Lakshitha and R. L. Dangalla, "A Novel Method for Freelancer Bids Ranking," in *Proc. 12th Int. Research Conf. (KDU IRC)*, Ratmalana, Sri Lanka, 2019.

[3] A. S. A. Fadzil, N. A. F. Othman, R. Thurasamy, and L. H. Johari, "Understanding the Bidding Process in Crowdsourcing Platforms among Freelancers," *EnvironmentBehaviour Proc. J.*, vol. 10, no. SI30, pp. 149–156, 2025.

[4] N. Joshi, S. Gupta, and R. K. Singh, "Dynamic Pricing in Freelancing: A Machine Learning Approach," in *Proc. Int. Conf. Adv. Comput. Res. Soc. (ICACRS)*, 2024, pp. 45–50.

[5] S. Kumar and P. Sharma, "Trust Mechanisms in Gig Economy Platforms," *Int. J. Eng. Res. Appl. (IJERA)*, vol. 14, no. 2, pp. 22–29, 2024.

[6] A. M. Alkhateeb, K. H. Hyari, and M. Hiyassat, "Analyzing bidding competitiveness and success rate of contractors competing for public construction projects," *Construction Innovation*, vol. 21, no. 4, pp. 576–591, 2020.

[7] V. Anand and C. Hota, "Motivation of participants in crowdsourcing platforms using intelligent agents," *International Journal of Computing*, pp. 78–87, 2020.

[8] E. Ayaburi, J. Lee, and M. Maasberg, "Understanding crowdsourcing contest fitness strategic decision factors and performance: an expectation-confirmation theory perspective," *Information Systems Frontiers*, vol. 22, no. 5, pp. 1227–1240, 2019.

[9] A. S. Bageis, I. I. Falqi, A. Alshehri, S. Alsulamy, and T. A. Alsahli, "Behavioral differences towards internal and external factors in making the bid/no bid decision," *Civil Engineering Journal*, vol. 5, no. 5, pp. 1189–1196, 2019.

[10] S. Barnes, A. E. Green, and M. D. Hoyos, "Crowdsourcing and work: individual factors and circumstances influencing employability," *New Technology, Work and Employment*, vol. 30, no. 1, pp. 16–31, 2015.