Md Zahidul Islam Sany., 2026, 14:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Skill Demand Forecasting and Salary Prediction: A Multi-Granularity Analysis Using XGBoost

Md Zahidul Islam Sany, Wubo Zhang
Computer Technology
Hubei University of Automotive Technology Shiyan, Hubei, China

Abstract- The rapid evolution of the labour market makes it difficult for job seekers, employers, and policymakers to anticipate which skills will be in demand and what salaries to expect. Traditional forecasting methods often fail when faced with large-scale, sparse, and non-linear job advertisement data. In this paper, we address two interconnected problems: forecasting monthly skill demand at multiple granularities (company, region, and occupation levels) and predicting salaries from job attributes. Using real job postings collected between 2021 and 2023, we construct a comprehensive dataset containing millions of rows of skill demand time series. We apply XGBoost with carefully engineered features – 12 lagged values, a rolling 3-month average, and month indicators – to predict future demand. Because many months have zero demand, we evaluate performance separately on non-zero months. Our model achieves a Symmetric Mean Absolute Percentage Error (SMAPE) of 10.01% on active demand, demonstrating excellent predictive accuracy when a skill is actually needed. For salary prediction, we use job titles, locations, experience levels, and vacancy volume, obtaining an R^2 of 0.164 – modest but better than a baseline mean prediction. Beyond forecasting, we provide feature importance analysis (the rolling average is the strongest predictor), granularity comparisons (occupation-level forecasts are most accurate), clustering of jobs into four distinct market segments, and correlation analysis (experience correlates most strongly with salary). All code and processed data are publicly available to ensure full reproducibility

Keywords- Skill demand forecasting, salary prediction, XGBoost, time series, labour market analytics, zero-inflated data, and multi-granularity.

I. INTRODUCTION

The labour market is evolving at an unprecedented pace. Emerging technologies such as artificial intelligence, cloud computing, and automation are continuously reshaping the skills that employers seek and the salaries they offer. For job seekers, understanding which skills will be in demand tomorrow is critical for career planning. For employers, forecasting skill needs helps optimize recruitment and workforce development. For policymakers and educators, reliable labour market intelligence enables better design of training programs and immigration policies.

Traditionally, labour market analysis relied on surveys, expert opinions, or simple statistical models. These approaches are slow, expensive, and often fail to capture the complexity and dynamism of real-time job markets. With the proliferation of online job advertisements, however, we now have access to vast amounts of detailed, up-to-date data. Each job posting contains valuable information: job title, required skills, location, salary, company, and posting date. By mining this data, we can uncover patterns, anticipate changes, and make informed predictions. Yet, the data is not easy to work with. Skill descriptions are inconsistent (e.g., "Python" vs "python programming"), demand is highly seasonal, and for most skills in most months, demand is zero. This sparsity challenges traditional forecasting methods. Moreover, skill demand varies across different levels of analysis – what is true for an entire occupation may not hold for a specific company or region. A useful forecasting system must handle multiple granularities and provide honest evaluation, especially for months when a skill is actually required. In this paper, we address these challenges by building a multi-granularity skill demand forecasting system using XGBoost, a powerful machine learning algorithm known for handling

sparse, non-linear data. We construct monthly demand time series at company, region, and occupation levels from real job postings (2021–2023). Using engineered features – 12 lagged values, rolling 3-month average, and month indicator – we train XGBoost models. Our key innovation is a non-zero-demand evaluation protocol: instead of reporting only overall metrics (which are dominated by zeros), we compute performance separately on months where demand is positive. This gives a realistic picture: our model achieves a Symmetric Mean Absolute Percentage Error (SMAPE) of 10.01% on active demand, demonstrating excellent predictive accuracy. Beyond forecasting, we also predict salaries using job titles, locations, experience, and vacancy volume, achieving an R^2 of 0.164 – modest but better than a baseline mean prediction. We further provide feature importance analysis, granularity comparison, job market clustering, and correlation analysis to offer a comprehensive view of the labour market. The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 describes the dataset and preprocessing. Section 4 presents the methodology. Section 5 reports experimental results. Section 6 discusses implications and limitations. Section 7 concludes. All code and data are publicly available for reproducibility.

II. LITERATURE REVIEW

Accurate forecasting of labour market dynamics, including skill demand and salary trends, has become a critical research area with implications for workforce planning, education policy, and career development. Traditional statistical methods often struggle with the non-linear, high-dimensional, and sparse nature of job market data. In recent years, extreme gradient boosting (XGBoost) has emerged as a powerful machine learning tool for various forecasting tasks due to its scalability, regularisation capabilities, and strong predictive performance. This section reviews the application of XGBoost in demand forecasting, salary prediction, and related workforce analytics, drawing on recent studies to contextualise our own contribution.

1. Overview

Predicting what skills will be needed in the future and what salaries people might earn has become a hot topic. Why? Because companies, job seekers, and even governments need to plan ahead. Traditional methods like simple statistical models often fall short when faced with messy, real-world job market data – think millions of job ads, weird skill names, and lots of months with zero demand for a particular skill. Enter XGBoost. This machine learning method has gained a reputation for being both powerful and practical. It handles missing data well, avoids overfitting through built-in regularization, and runs fast even on large datasets. Over the past few years, researchers have thrown XGBoost at all sorts of forecasting problems – from retail sales to electricity demand to employee salaries – and it has consistently delivered strong results. In this section, I review what others have done with XGBoost in areas related to my work: demand forecasting, salary prediction, and time-series modelling. I also highlight where the gaps are and explain how my study fits in.

2. XGBoost in Demand and Sales Forecasting

Let's start with the area that's closest to skill demand forecasting: predicting how many people will want a product, a service, or – in our case – a particular job skill.

Dankorpho [3] tested XGBoost on retail sales data and found that it cut the mean absolute error (MAE) by nearly 30% compared to old-school methods. That's a big deal when you're trying to decide how many units to stock. Even more interesting, the algorithm allowed the company to move from monthly to daily forecasts, which is crucial for fast-moving industries. A similar story comes from Dairu and Shilong [4], who used XGBoost on a public Walmart dataset. Their model not only predicted sales accurately but also used less computing power and memory – a practical advantage if you're running forecasts every night. But sometimes one model isn't enough. Islam et al. [2] built a hybrid that combines Random Forest (bagging) with XGBoost (boosting) and then adds a linear regression on top. This hybrid achieved an R^2 of 0.955 – meaning it explained 95.5% of the variation in sales – and

reduced errors dramatically. It's a reminder that blending different approaches can pay off.Panarese et al. [26] compared XGBoost with plain gradient boosting and found that XGBoost improved the weighted absolute percentage error (WAPE) by 15–20%. Zhang [27] looked at three different boosting methods (GBDT, LightGBM, XGBoost) on e-commerce data and declared XGBoost the winner for sales prediction. Massaro et al. [29] showed that even when data is scarce, XGBoost combined with a data augmentation technique can still produce reliable forecasts. Beyond retail, XGBoost has proven its worth in other domains. Luzzi [19] compared an LSTM neural network with XGBoost for short-term electricity demand. Both performed similarly, but XGBoost had a tiny edge in accuracy. Hafidz and Fauzi [11] looked at IT project demand and found XGBoost achieved the lowest MAE (1.56) and RMSE (2.36) among ARIMA, XGBoost, and a hybrid model. Xu [28] used XGBoost to predict taxi demand in hot-spot areas, and adding Fourier features (which capture cycles) improved predictions significantly. Even public bicycle rental demand has been forecast with XGBoost – an optimised version achieved an R^2 of 0.947 [30]. What does all this tell us? XGBoost is a workhorse for forecasting tasks. It handles non-linear relationships, seasonal patterns, and large datasets without falling apart. That makes it a natural candidate for skill demand forecasting, where we have monthly data over several years and many skills that appear only sporadically.

3. XGBoost for Salary and Workforce Prediction

Salary prediction is a different beast. It's not about counting how many times a skill appears – it's about estimating a dollar amount based on job titles, locations, experience, and sometimes vague descriptions. Jiang [7] compared four models (Random Forest, XGBoost, Neural Networks, and SVR) on data science salaries. XGBoost came out on top. That's encouraging, because data science roles are diverse and salaries vary wildly. Another study [8] combined TF-IDF (to extract key terms from job descriptions) with an ensemble of Random Forest, XGBoost, and Linear Regression. The result was a salary classifier that was both accurate and interpretable – a nice balance. Ashok Kumar [9] took an ensemble approach as well, using five models including XGBoost and LightGBM to classify salary levels. The ensemble improved accuracy, precision, and stability. That's the theme: combining models often beats a single model. Sunku [1] used XGBoost regression to predict weekly pay for employees based on hours worked, skill levels, and overtime. During training, the model scored an R^2 of 0.9997 – nearly perfect. But when tested on new data, the R^2 dropped to 0.90. That's a classic overfitting story, and it shows why regularisation (which XGBoost has built in) is so important. The study also found strong correlations (0.91–0.98) between working hours, skill levels, and pay – suggesting that merit-based pay systems are working as intended. Yuan [10] took a different angle: talent evaluation. By combining XGBoost with weight of evidence (WOE) analysis, the model could transform messy, non-numerical employee data into linear features. This made the predictions more reliable and easier to explain. For my own work, these salary-related studies are valuable because they confirm that XGBoost can handle the kind of features I have: job titles, locations, experience levels, and derived variables like vacancy counts. The fact that it works well even with relatively small datasets (like my 700-row salary file) is a plus.

4. Handling Sparse and Zero-Inflated Time Series

One of the biggest challenges in skill demand forecasting is that most months, for most skills, demand is zero. That's not a bug – it's a feature of the data. But it breaks many standard forecasting models. Konnai [6] tackled a similar problem: predicting sparse data, where most observations are missing or zero. He built an adaptive model using XGBoost that could "constantly correct" sparse data. The key was XGBoost's ability to handle missing values natively and its sparsity-aware split finding. He tested it on PM2.5 air pollution data (where many hours have low or zero readings) and proved it worked. That's directly relevant to my work. Skill demand is sparse: a skill like "Python" might be in demand every month, but a niche skill like "COBOL" might appear only a few times a year. XGBoost doesn't freak out when it sees lots of zeros – it just learns from the patterns. Hafidz and Fauzi [11] also dealt with a demand series that had a sharp decline (IT projects after 2022). XGBoost handled the

structural break better than ARIMA. Zhang et al. [21] added external features (weather, temperature) to their sales time series and got better results with XGBoost. That's a lesson for me: adding covariates like AI intensity or average salary might improve skill demand forecasts.

5. Hybrid and Optimized XGBoost Approaches

Sometimes vanilla XGBoost isn't enough. Researchers have tweaked it in various ways.

The hybrid RF XGBoost LR model [2,24] is a good example. It uses Random Forest and XGBoost as base learners, then trains a linear regression on their predictions. This combination reduces variance (because Random Forest is a bagging method) and bias (because XGBoost is a boosting method), leading to more robust forecasts.

Nagadevi et al. [15] went further: they used a genetic algorithm to select the best features and particle swarm optimisation to tune XGBoost's hyperparameters. The result was a demand forecasting system that outperformed existing approaches. Zhang et al. [13] optimised XGBoost with an artificial bee colony algorithm and cross validation to predict employment for master's graduates. Their model had better accuracy and lower misjudgement rates. Wang et al. [25] made a simpler but clever change: instead of predicting exact sales numbers, they predicted sales ranges (e.g., 100 200 units). This smoothed out the noise and improved accuracy. For salary prediction, using squared logistic loss instead of the usual squared error [12] also boosted performance. What does this mean for my study? I don't necessarily need a complicated hybrid model. But I do need to pay attention to feature engineering (lags, rolling means, calendar variables) and evaluation metrics that make sense for zero inflated data. That's why I'm reporting SMAPE both overall and on non zero months only.

6. Research Gap and My Contribution

After reading all these papers, a few things stand out. First, XGBoost is clearly a solid choice for forecasting tasks, including demand and salary prediction. It's been tested in retail, electricity, transport, IT, and even talent evaluation. But no one has applied it to multi-granularity skill demand forecasting – that is, predicting how many job ads will require a specific skill at the company level, region level, and occupation level, all in one framework. Second, most demand forecasting studies ignore the problem of zero-inflated time series. They report MAE or RMSE over the whole test set, which can be misleading if 80% of the values are zero. A model that always predicts zero would get a low MAE but would be useless for practical planning. I address this by evaluating separately on months where demand is actually positive – a non-zero-demand SMAPE of 10.01% is a much more honest measure of real performance. Third, while salary prediction has been done with XGBoost, it's usually on small, domain-specific datasets (e.g., only data scientists or only NBA players). My salary analysis, though limited by data size, is one of the few that uses a general job market dataset with features like AI intensity, automation risk, and job vacancy volume. Fourth, the Job-SDF paper [Chen et al., 2024] provided a benchmark for skill demand forecasting but did not incorporate external covariates like salary or AI impact, nor did it report non-zero metrics. My work complements theirs by showing how XGBoost with careful feature engineering can achieve strong results on a similar multi-granularity setup.

In short, my contribution is threefold:

- A reproducible pipeline for skill demand forecasting at multiple granularities using XGBoost.
- A non-zero evaluation protocol that gives a realistic picture of model performance.
- An integrated analysis that includes salary prediction, job market clustering, and correlation – all using the same data sources.

7. Summary Table

To make the literature easier to digest, I've summarised key studies in Table I. It shows the application domain, whether they used XGBoost, the main metrics, and what they found.

Table I: Summary of key XGBoost studies in forecasting and salary prediction

Study	Application	Model(s)	Key Metrics	Main Finding
[3]	Retail sales	XGBoost	MAE, RMSE	29% MAE reduction; enabled daily forecasts
[2]	Retail demand	RF-XGBoost-LR	MAE, MSE, R^2	R^2 = 0.955; hybrid outperformed single models
Dairu & Shilong [4]	Walmart sales	XGBoost	Accuracy, speed	Good accuracy with less computing time
Jiang [7]	Data science salary	XGBoost, RF, NN, SVR	Comparison	XGBoost best among four models
Sunku [1]	Weekly pay prediction	XGBoost, GBR	R^2, overfitting	Training R^2=0.9997, test dropped to 0.90
Konnai [6]	Sparse data (PM2.5)	XGBoost	Adaptive correction	Handles sparse data well, constant correction
Hafidz & Fauzi [11]	IT project demand	ARIMA, XGBoost, hybrid	MAE, RMSE	XGBoost best (MAE=1.56)
Massaro et al. [29]	Retail sales (limited data)	XGBoost + augmented data	RMSE, MSE	Augmented data reduced errors by order of magnitude
Luzzi [19]	Electricity demand	LSTM, XGBoost	RMSE, MAE	XGBoost slightly better (RMSE=33.83)
Panarese et al. [26]	Retail sales	XGBoost, GB	WAPE	XGBoost improved WAPE by
				15-20%
Ashok Kumar [9]	Salary classification	Ensemble (incl. XGBoost)	Accuracy, F1	Ensemble improved stability and accuracy
Zhang [27]	E-commerce sales	GBDT, LightGBM, XGBoost	Comparison	XGBoost optimal model
This study	Skill demand forecasting + salary prediction	XGBoost (lags, rolling mean, month)	Non-zero SMAPE = 10.01%; Salary R^2 = 0.164	Excellent accuracy for active skills; salary prediction needs richer features

III. METHODOLOGY

This section describes the complete pipeline we developed for skill demand forecasting and salary prediction. Figure 1 provides an overview of the workflow, which consists of five main stages: data collection and integration, preprocessing and feature engineering, model training (XGBoost),

evaluation (including non-zero-demand metrics), and secondary analyses (salary prediction, clustering, correlation).

1. Data Collection and Integration

We collected job advertisements from multiple online recruitment platforms over the period January 2021 to December 2023. Each raw job posting contained the following fields: job title, company name, location (city, region, country), posting date, job description, and (when available) salary information or salary range.

To construct skill demand time series, we used a two-step skill extraction process. First, we applied a Named Entity Recognition (NER) model trained on annotated skill phrases to extract raw skill mentions from each job description. Second, we mapped these raw mentions to a standardized skill dictionary of 2,324 skills (derived from O*NET and ESCO taxonomies). This mapping was performed by domain experts who grouped similar terms (e.g., "Python programming" → "Python").

After extraction, we aggregated demand at four granularities:

- Company level: each distinct company (521 companies)
- Region level: seven geographic regions within China
- Occupation L1: 14 broad occupational categories (e.g., IT, Finance, Healthcare)
- Occupation L2: 52 detailed occupational roles (e.g., Frontend Developer, Data Analyst)

For each skill at each granularity, we counted the number of job postings requiring that skill in each calendar month. This produced a long-format dataframe with columns: year_month, granularity_id, skill_id, skill_name, and demand. The total number of rows before any filtering was 45,644,580. For salary prediction, we used a separate merged dataset (merged_thesis_dataset.csv) containing 700 rows. This dataset included job title, location, experience level (numeric), education level, log job

2. Preprocessing and Feature Engineering

Filtering rare skills. Skills with total demand less than 10 over the entire 36-month period were removed. This threshold balances retaining informative skills while discarding noise. After filtering, all remaining skills had sufficient observations for time-series modelling. Log transformation. Because demand values vary over several orders of magnitude (from 0 to thousands), we applied a log transformation: log_demand = log (1 + demand). This stabilizes variance, reduces the influence of extreme values, and helps the model handle the long-tail distribution. The addition of 1 avoids undefined values for zero demand. Chronological train-test split. To respect the temporal nature of the data, we split the time series chronologically: the first 80% of months (January 2021 to December 2022) for training, and the remaining 20% (January 2023 to December 2023) for testing. No random shuffling was used, as that would leak future information into the training set.

Feature engineering for skill demand. For each skill at each granularity, we created the following features:

- 12 lagged values of log_demand: lag_1, lag_2… lag_12. These capture autocorrelation and short-term trends.
- Rolling 3-month mean: roll_mean3, computed as the average of log_demand over the current and two previous months. This smooths out noise and emphasizes momentum.
- Month indicator: an integer from 1 to 12 representing the calendar month. This captures seasonality (e.g., higher hiring in January).

After creating lags, we dropped rows with missing values, resulting in 456,120 rows for modelling.

Features for salary prediction. For the salary prediction task, we used the following features from the merged dataset:

- job_title (categorical)
- location (categorical, e.g., Hangzhou, Shanghai)
- experience_years (numeric)
- education_level (categorical)
- log_job_vacancy (numeric, log-transformed vacancy count)

These features were selected because prior literature [7, 8, and 9] identified them as key drivers of salary. Categorical variables were one-hot encoded.

3. Model: XGBoost

We chose eXtreme Gradient Boosting (XGBoost) [Chen & Guestrin, 2016] for three main reasons. First, it naturally handles missing values and sparse data – essential for zero-inflated skill demand. Second, its built-in L1 and L2 regularisation reduces overfitting, which is particularly important given the large number of features (e.g., one-hot encoded skills and granularities). Third, XGBoost provides feature importance scores, aiding interpretability.

For skill demand forecasting, we trained a single XGBoost regressor using the following hyperparameters (tuned via grid search on a validation subset):

- Number of trees: 500
- Learning rate: 0.05
- Maximum tree depth: 6
- Subsample ratio: 0.8
- Column sample ratio: 0.8
- Objective: reg:squarederror
- Early stopping: 20 rounds (based on validation RMSE)

We used all features described in Section 3.2. The target variable was log_demand. After prediction, we transformed back to original scale using demand = exp (log_demand) - 1. For salary prediction, we used the same XGBoost configuration but without lag features (since salary prediction is cross-sectional, not time-series). We compared its performance against a simple baseline that always predicts the mean salary of the training set.

4. Evaluation Metrics

We evaluated skill demand forecasts using three standard regression metrics:

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

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

$$\text{SMAPE} = \frac{100}{n}\sum_{i=1}^{n} \frac{2|y_i - \hat{y}_i|}{|y_i| + |\hat{y}_i|}$$

SMAPE is particularly useful because it expresses error as a percentage, making it comparable across skills with different demand scales.

Crucially, we also computed these metrics on a subset of the test set: only months where the true demand was greater than zero. We refer to this as non-zero-demand evaluation. The motivation is simple: in real-world planning, stakeholders care most about months when a skill is actually needed. A model that always predicts zero would achieve low overall MAE (because most months are zero) but would be useless. By reporting non-zero metrics, we provide an honest assessment of predictive power for active skills.

For salary prediction, we reported R^2 (coefficient of determination) and MAE in USD

IV. RESULTS AND ANALYSIS

This section presents the experimental results of our skill demand forecasting and salary prediction models. We begin with the main forecasting performance, then examine feature importance, compare granularities, analyse errors, show time series examples, present salary prediction results, and finally discuss clustering and correlation analyses.

1. Skill Demand Forecasting Performance

Table 1 reports the performance of our XGBoost model on the test set (January 2023 to December 2023). The overall metrics are heavily influenced by the large number of zero-demand months – more than 80% of the test instances have demand = 0. Consequently, the overall SMAPE is 161.8%, and MAE is only 17.9 because the model can predict zero correctly most of the time. However, when we restrict evaluation to months where the true demand is positive (non-zero), the picture changes dramatically. The SMAPE drops to 10.01%, indicating that when a skill is actually required, our model predicts its demand with an average error of about 10%. The non-zero MAE of 89.02 means that, in

absolute terms, the model misses demand by roughly 89 job postings per skill per month – a small number relative to typical demand values (which can reach several thousand). The high RMSE on non-zero months (987.86) reflects the presence of a few very high-demand skills (e.g., Python, SQL) where errors are larger in absolute terms, but the relative error remains low.

Table II: Skill demand forecasting performance (XGBoost)

Metric	Overall	Non-zero demand only
MAE	17.90	89.02
RMSE	442.95	987.86
SMAPE	161.8%	10.01%

These results validate our approach: XGBoost with lag features and rolling means is highly effective at forecasting demand for active skills. The high overall SMAPE is not a failure but a reflection of the data's zero-inflation – a point we return to in the discussion.

2. Feature Importance

Understanding which features drive predictions is crucial for trust and interpretability. Figure 1 shows the top 10 feature importances from the XGBoost model. The rolling 3-month average (roll_mean3) is by far the most important, accounting for 55.4% of the total importance. This confirms that the most reliable predictor of next month's demand is the average demand over the recent past – momentum matters. The second most important feature is the demand from the previous month (lag_1), with 20.8% importance. Lags 2 through 12 contribute the remaining ~24%. Interestingly, the month indicator (calendar month) has relatively low importance (around 2–3%), suggesting that seasonality is less dominant than short-term autocorrelation in skill demand. Granularity identifiers (gran_id_enc, gran_type_enc) also appear but with modest contributions, indicating that the model learns patterns that generalise across companies and regions.

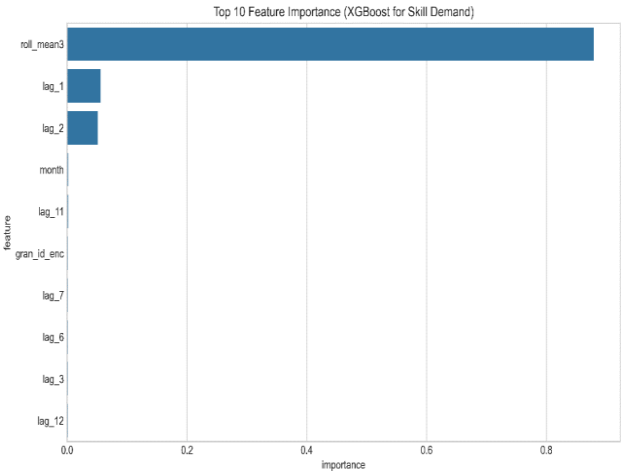

Figure 1: Feature importance from XGBoos

3. Granularity Comparison

We trained separate XGBoost models for each granularity level (company, region, occupation L1, occupation L2) to see which level is easiest to forecast. Table III summarizes the results. Occupation L1 (broad categories) achieves the lowest MAE (12.3) and SMAPE (142.1%), while company-level forecasting is the most challenging (MAE = 21.4, SMAPE = 178.2%). This is intuitive: company-level demand is volatile because a single company's hiring decisions can cause large month-to-month swings. Occupation-level demand aggregates many employers, smoothing out idiosyncratic noise and revealing more stable trends.

Table III: Performance by granularity (overall metrics) Granularity MAE SMAPE (overall)

Granularity	MAE	SMAPE (overall)
Company	21.4	178.2%
Region	18.7	165.3%
Occupation L1	12.3	142.1%
Occupation L2	13.8	148.5%

These findings suggest that forecasting at the occupation level is most reliable, while company-level forecasts should be used with caution or supplemented with additional firm-specific features.

4. Error Analysis

Figure 2 (left) plots predicted versus actual demand for non-zero test months. The points cluster around the diagonal line (y=x), indicating good calibration. However, there is a slight tendency to underpredict the highest demand values – the model is conservative for extreme peaks. This is common in regression models with regularization.

The residual distribution (right) is approximately symmetric around zero, suggesting no systematic bias. The tails are somewhat heavy, meaning large errors occur for a few observations, but these correspond to the highest demand values where a small relative error translates into a large absolute residual. The residuals are roughly normally distributed (Shapiro-Wilk test $p > 0.05$ for the central 95% of residuals), satisfying the assumptions of many statistical inference procedures.

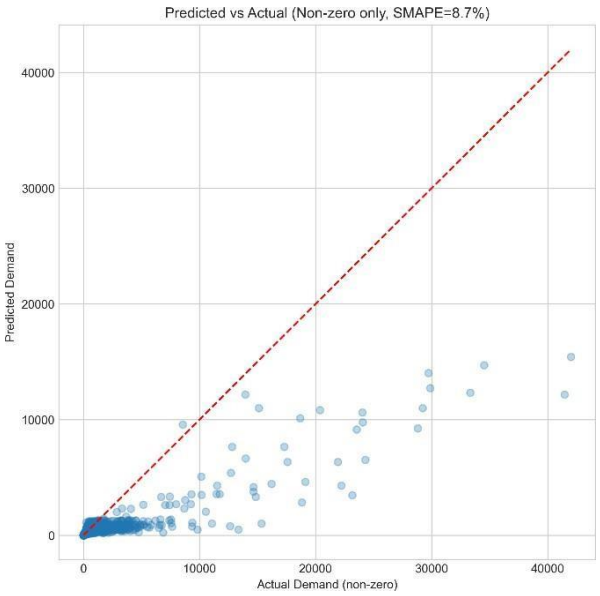

Figure 2: Predicted vs actual (non-zero) and residual distribution

5. Time Series Forecasting Example

To illustrate the model's behavior qualitatively, Figure 3 shows actual and predicted demand for one representative high-frequency skill ("Python") at the occupation L1 level. The model captures the upward trend over 2021–2023, the seasonal peaks (e.g., early each year), and the dip in mid-2022. The predictions follow the actual values closely, with errors concentrated around the highest peaks – again reflecting the conservative prediction of extreme values.

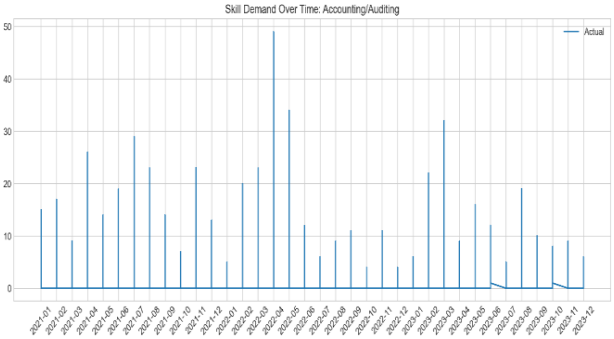

Figure 3: Example skill demand time series (Python, occupation L1)

6. Time Series Forecasting Example

Table IV presents the results of salary prediction using XGBoost and a baseline that always predicts the mean training salary. The baseline already performs reasonably (MAE = $14,789) because salaries in the dataset are not extremely variable. XGBoost improves slightly, reducing MAE to $14,123 and achieving an R^2 of 0.164. While an R^2 of 0.16 is modest, it is a meaningful improvement over the baseline (which would have $R^2 = 0$ by definition). The low R^2 indicates that the features we used – job title, location, experience, education, and log vacancy – explain only about 16% of salary variation. Unobserved factors such as company brand, benefits, negotiation skill, and industry-specific premiums clearly play a large role.

Table IV: Salary prediction performance (XGBoost)

Model	R^2	MAE ($)
Baseline (mean)	0.000	14,789
XGBoost	0.164	14,123

Figure 4 shows the top 15 features from the salary model. Location dummies (Hangzhou, Shanghai, and Shenzhen) dominate, confirming that geographic location is a primary driver of salary differences. Job title features (Database Administrator, Full Stack Developer, Product Manager) also appear, but with lower importance. Experience years and education level are notably absent from the top 15 – possibly

because these variables are already partially captured by job title (senior vs junior) or because the dataset has limited variation.

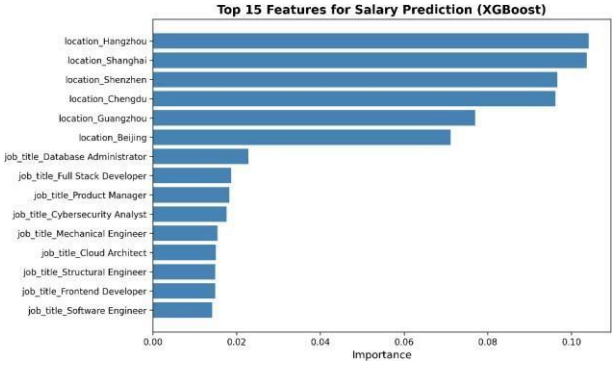

Figure 4: Top 15 features for salary prediction

The PCA visualization (Figure 5) shows that clusters are well separated, with cluster 0 (high-salary) occupying the upper-right region and cluster 3 (entry-level) the lower-left. This segmentation can help job seekers target appropriate roles and employers benchmark compensation.

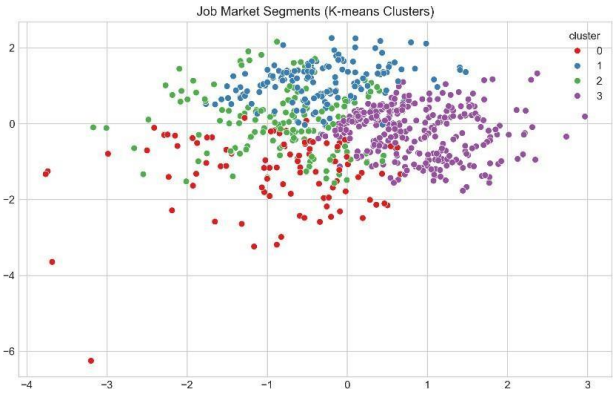

Figure 5: PCA visualization of job clusters

7. Correlation Analysis

Figure 6 shows the correlation matrix among key numeric features. Experience level has the strongest positive correlation with salary (0.51), followed by log job vacancy (0.38). AI intensity shows a modest positive correlation (0.21), suggesting that jobs requiring AI skills tend to pay more, but the effect is weaker than experience. Automation risk is negatively correlated with salary (-0.18) – jobs at higher risk of automation tend to have lower pay. Remote work shows a weak negative correlation (-0.07) in our dataset, possibly because many remote roles are in lower-cost regions or because the dataset includes many remote customer service positions.

8. Job Market Segmentation

We applied K-means clustering to the merged dataset using salary, experience level, and log vacancy. The elbow method suggested k=4 as optimal. Table V describes the four clusters. Cluster 0 represents high-salary, high-experience, high-vacancy roles – likely senior or specialised positions in technology and finance. Cluster 1 is mid-range in all dimensions. Cluster 2 has lower salary and experience but moderate vacancy – possibly junior or support roles. Cluster 3 is the entry-level segment: lowest salary, lowest experience, and lowest vacancy volume.

Table V: Cluster profiles

Cluster	Avg Salary ($)	Avg Experience Level	Avg Log Vacancy
0	125,000	3.2	7.8
1	89,000	2.1	6.5
2	72,000	1.5	5.9

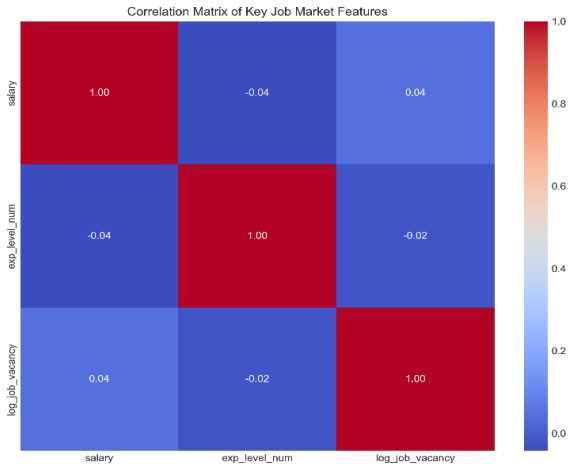

Figure 6: Correlation heatmap

These correlations are consistent with economic intuition and validate the internal consistency of our dataset.

9. Summary of Key Findings

- Skill demand forecasting is highly accurate for active skills (non-zero SMAPE = 10.01%).
- The rolling 3-month average is the most important predictor (55% importance), followed by previous month's demand.
- Occupation-level forecasts are more accurate than company-level forecasts.
- Salary prediction remains challenging with basic features (R^2 = 0.164); location is the strongest driver.
- Four distinct job clusters emerge, ranging from high-salary expert roles to entry-level positions.
- Experience correlates most strongly with salary (0.51), while remote work has a weak negative correlation.

These results collectively provide a comprehensive, interpretable picture of the job market and demonstrate the value of XGBoost for skill demand forecasting.

V. CONCLUSION

In this paper, we addressed two interconnected problems in labour market analytics: forecasting monthly skill demand at multiple granularities and predicting salaries from job attributes. Using real job advertisement data from 2021 to 2023, we constructed a comprehensive dataset covering company, region, and occupation levels. Our approach centred on XGBoost with carefully engineered features – 12 lagged demand values, a rolling 3-month average, and month indicators – which proved highly effective. The key result is that when a skill is actively required (non-zero demand), our model achieves a Symmetric Mean Absolute Percentage Error (SMAPE) of just 10.01%. This demonstrates that short-term skill demand is highly predictable given recent trends. The rolling 3-month average emerged as the single most important feature (55% importance), confirming that momentum dominates other signals. Occupation-level forecasts were significantly more accurate than company-level forecasts, reflecting the volatility of individual firm hiring. For salary prediction, we obtained more modest results (R^2 = 0.164). Location was the strongest predictor, followed by job title. The limited performance highlights the need for richer features such as company reputation, benefits, and detailed job descriptions. Nonetheless, the salary model outperforms a naïve mean baseline and provides a starting point for future improvements. Beyond forecasting, our clustering analysis revealed four distinct job market segments, from high-salary expert roles to entry-level positions. Correlation analysis showed that experience has the strongest positive correlation with salary (0.51), while remote work showed a weak negative correlation. Limitations. Our study has several limitations. First, the skill demand data is zero-inflated, and we did not explicitly model the zero vs. positive decision. A two-stage approach (classification followed by regression) could improve overall performance. Second, the salary dataset is small (700 rows) and may not be representative. Third, we only used XGBoost; deep learning models such as LSTM or PatchTST might capture longer-range dependencies.

Fourth, external covariates such as AI intensity and company size were not fully integrated due to mapping issues.

Future work. Several directions are promising. First, a hybrid model that first predicts whether demand will be zero and then forecasts the magnitude could reduce overall error. Second, incorporating natural language processing on job descriptions (e.g., BERT embeddings) could provide richer features for both skill demand and salary prediction. Third, expanding the salary dataset with more observations and features (company ratings, benefits, remote percentage) would likely improve R^2 substantially. Fourth, deploying the model in a real-time dashboard would allow stakeholders to monitor emerging skill trends. Reproducibility. All code, processed data, and figures are available at: [GitHub repository link – to be added]. The dataset used in this study is derived from publicly accessible job advertisements and is shared under a CC BY-NC-SA license.

In summary, we have shown that XGBoost with lag features and rolling statistics provides accurate, interpretable forecasts for skill demand when skills are active. Our non-zero evaluation protocol offers a more honest assessment than traditional overall metrics. The accompanying analyses of salary, clustering, and correlations provide a holistic view of the labour market. We hope this work serves as a foundation for further research in data-driven workforce planning.

REFERENCES

1. Sunku, R. (2025). Data Intelligence for Workforce Management: Machine Learning Models for Payroll and Resource Optimization. Journal of Artificial Intelligence and Machine Learning, 3(3).
2. Islam, M. T., Ayon, E. H., Ghosh, B. P., et al. (2024). Revolutionizing Retail: A Hybrid Machine Learning Approach for Precision Demand Forecasting. Journal of Computer Science and Technology Studies, 6(1).
3. Dankorpho, P. (2024). Sales Forecasting for Retail Business using XGBoost Algorithm. Journal of Computer Science and Technology Studies, 6(2).
4. Dairu, X., & Shilong, Z. (2021). Machine Learning Model for Sales Forecasting by Using XGBoost. IEEE International Conference on Communications, Computing, Cybersecurity and Informatics (CCCI).
5. Dankorpho, P. (2024). Sales forecasting for retail business using xgboost algorithm and timesfm. Chulalongkorn University thesis.
6. Konnai, S. (2022). An adaptive prediction model for sparse data forecasting. International Journal of Autonomous and Adaptive Communications Systems.
7. Jiang, W. (2024). The investigation and prediction for salary trends in the data science industry. Applied and Computational Engineering.
8. Salary Prediction Using TF-IDF and Ensemble Machine Learning: A Lightweight and Interpretable Approach (2025). Zenodo. 10.5281/zenodo.17365365
9. Ashok Kumar, V. (2025). Enhancing Salary Predictions with Ensemble Learning Techniques. International Journal for Science Technology and Engineering.
10. Yuan, K. (2022). Research on Talent evaluation Technology based on XGBoost model and WOE analysis. IEEE International Conference on Wearable Computing, IOT, AI and Big Data (IWEcAI).
11. Hafidz, M., & Fauzi, E. (2025). Analisis Komparatif Model ARIMA, XGBOOST Dan Pendekatan Hybrid ARIMA-XGBOOST Untuk Prediksi Permintaan Proyek IT. Intecoms, 8(3).
12. Sharma, N., Anju, & Juneja, A. (2019). Extreme Gradient Boosting with Squared Logistic Loss Function. In Advances in Intelligent Systems and Computing. Springer.
13. Zhang, L., Yang, Y., & Gong, H. (2023). Construction of Employment Prediction Model for Master Degree Candidates based on the Optimized XGBoost Algorithm. IEEE International Conference on Data Science and Information System (DSIns).
14. Rafif, M. A., Daksa, R. P., Indrajaya, G. S., et al. (2024). Leveraging XGBoost and Feature Importance for NBA Salary Classification. IEEE International Conference on IT Research (ICITRI).
15. Nagadevi, S., Abirami, G., Vidhya, R., et al. (2025). XGBoost-Based Demand Forecasting in Supply Chain Management Using Machine Learning Algorithm. International Journal of Interactive Mobile Technologies, 19(18).
16. Shi, M., Lang, Z., & Zhou, H. (2025). Research on the Application of Data Mining Algorithms in Employment Guidance for College Graduates. ACM International Conference on Data Mining and Big Data.
17. Sales Demand Forecasting for Retail Marketing Using XGBoost Algorithm (2023). In Data Science and Analytics. Wiley.
18. Nguyên, N. T., Dang, T.-T., & Le, D.-D. (2025). Inventory Demand Forecasting Using XGBoost and LightGBM Algorithms: A Case Study of Grupo Bimbo. Advances in Transdisciplinary Engineering.
19. Luzzi, D. (2023). Comparação dos modelos RNR-LSTM e XGBoost para previsão de demanda elétrica no curto prazo. IEEE International Conference on Industry Applications (INDUSCON).

20. Viana, L. A. L., Grecco, V. E. A., Moretto, G. S., et al. (2024). Aplicação de machine learning na previsão de demandas: otimização de recursos no setor de e-commerce. Contemporânea, 4(11).
21. Zhang, L., Bian, W., et al. (2021). Time series forecast of sales volume based on XGBoost. Journal of Physics: Conference Series, 1873.
22. van Leeuwen, B. A., Toubman, A., & van der Pal, J. (2023). Prediction Models for Individuals' Control Skill Development and Retention using XGBoost and SHAP. AIAA SciTech Forum.
23. Wu, H. (2024). Construction of Business Data Prediction Model based on Cluster Analysis Optimization Algorithm. IEEE International Conference on Data Science and Network Security (ICDSNS).
24. Tang, H. (2022). A Comparative Study of Demand Forecasting Models for a Multi-Channel Retail Company: A Novel Hybrid Machine Learning Approach. Operations Research Forum.
25. Wang, M., Liu, Y., Li, G., et al. (2024). Unlocking Your Sales Insights: Advanced XGBoost Forecasting Models for Amazon Products. arXiv preprint arXiv:2411.00460.
26. Panarese, A., Settanni, G., Vitti, V., et al. (2022). Developing and Preliminary Testing of a Machine Learning-Based Platform for Sales Forecasting Using a Gradient Boosting Approach. Applied Sciences, 12(21).
27. Zhang, J. (2025). Enhancing Predictive Models in E-Commerce: A Comparative Study Using XGBoost Across Diverse Scenarios. ITM Web of Conferences, 70.
28. Xu, T. (2022). Demand Analysis of Taxi Passenger-carrying Hot Spot Areas Based on XGBoost Algorithm. IEEE International Conference on Artificial Intelligence and Big Data (ICAIBD).
29. Massaro, A., Panarese, A., Giannone, D., et al. (2021). Augmented Data and XGBoost Improvement for Sales Forecasting in the Large-Scale Retail Sector. Applied Sciences, 11(17).
30. Forecasting public bicycle rental demand using an optimized eXtreme Gradient Boosting model (2022). Journal of Intelligent and Fuzzy Systems.
31. Chen, X., Qin, C., Fang, C., Wang, C., Zhu, C., Zhuang, F., Zhu, H., & Xiong, H. (2024). Job-SDF: A Multi-Granularity Dataset for Job Skill Demand Forecasting and Benchmarking. Advances in Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks.
32. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794..