

Emerging IT Skill Demand Analysis and Skill Trend Prediction Using Job Posting Data and Machine Learning

Dhanya Sri G¹, Bhuvaneswari K², Kavya M³

M.Sc. Student, Computer Science (Data Analytics), Dr. N.G.P. Arts and Science College, Coimbatore, India¹

M.Sc. Student, Computer Science (Data Analytics), Dr. N.G.P. Arts and Science College, Coimbatore, India²

M.Sc. Student, Computer Science (Data Analytics), Dr. N.G.P. Arts and Science College, Coimbatore, India³

Abstract: Online job advertisements describe the skills employers request, but this information is embedded in unstructured text. This paper presents a data analytics framework that cleans job-posting records, extracts IT skills with a documented dictionary, measures demand as the share of unique postings mentioning each skill, identifies emerging skills from changes in that share, and tests whether machine learning can forecast next-month demand better than simple baselines. The framework is implemented and validated on a simulated corpus of 11,820 postings spanning 36 months (January 2023 to December 2025) with 30 skills. Rule-based extraction with context checks removed all 269 false matches produced by naive keyword matching on this corpus. In rolling one-step-ahead forecasting over a nine-month test period, a damped-trend exponential smoothing model gave the lowest error (MAE 1.49 percentage points), 18.0% lower than the naive baseline, while a Random Forest on lag features did not outperform a moving average. Because the corpus is simulated, the numerical rankings and trends demonstrate the method only and are not measurements of the real labour market; the same code is intended to be re-run on a documented public job-posting dataset.

Keywords: IT skills, job postings, text mining, skill extraction, machine learning, demand forecasting.

I. INTRODUCTION

Digital recruitment platforms publish job advertisements that describe responsibilities, qualifications, and expected skills. Analysed together, these advertisements show which competencies employers mention when recruiting, and how those mentions change over time. Such evidence is useful to students choosing what to learn, to colleges revising curricula, and to early-career professionals planning their development. A vacancy is, however, an advertisement and not a complete record of hiring; any finding is a statement about advertised demand within the dataset analysed.

Skill information is usually embedded in free text. The same skill appears under several names (for example “k8s” and “Kubernetes”), a skill may be repeated within one advertisement, the same advertisement may be posted several times, and some skill names collide with ordinary words. Without consistent cleaning, alias normalization, and duplicate handling, apparent demand is distorted. Earlier work has explored job-ad classification, skill extraction, and the use of vacancy text for labour-market intelligence [1]–[4].

This paper contributes (i) an end-to-end, reproducible workflow from raw postings to skill-share time series, (ii) an evaluation of rule-based skill extraction against keyword matching, (iii) a simple definition of emerging skills based on relative growth in share, and (iv) a time-ordered comparison of forecasting methods against naive baselines. The workflow is demonstrated on a simulated posting corpus so that every step can be verified against known ground truth.

II. RELATED WORK

Research on machine-learning classification of online job advertisements shows that vacancy text can be converted into structured labour-market information, although results depend on the source, language, and classification taxonomy [1]. Studies of AI-related job postings have examined skill bundles and shown how posting text can reveal combinations of technical skills [2]. Research on Indian online job advertisements has used domain-specific word embeddings to categorize skills, which is relevant to Indian job markets [3]. A recent survey reviews deep-learning approaches to skill extraction and classification and highlights terminology variation and data quality as

persistent challenges [4]. Other work compares skill-categorization methods for job advertisements using wage regressions [5].

For forecasting, damped-trend exponential smoothing is a standard method for series with a trend that may flatten [9], and textbooks recommend comparing any model with simple baselines on time-ordered test data [8]. Tree ensembles such as Random Forests [7] are widely used for tabular prediction but average training observations, which limits extrapolation beyond the range seen in training. These studies motivate a transparent workflow that links extraction, frequency analysis, time aggregation, and baseline-controlled forecasting.

III. PROBLEM STATEMENT AND OBJECTIVES

Job advertisements contain useful information about IT skills, but unstructured wording and inconsistent terminology make comparison difficult. Repeated postings, uneven coverage, and changing posting volume can distort apparent demand. The objectives of this study are:

- 1)To clean and standardize job-posting records, removing missing, invalid, and duplicate entries.
- 2)To extract IT skills using a documented dictionary with alias normalization and context rules, and to measure extraction quality.
- 3)To measure skill frequency and co-occurrence using unique postings as the unit of count.
- 4)To identify emerging skills from changes in monthly skill share.
- 5)To forecast monthly skill share and compare machine-learning models with simple baselines using time-ordered evaluation.

IV. METHODOLOGY

A. Dataset

A real job-posting extract would require a source with clear reuse terms, and its origin, download date, licence, record count, and date range would be reported. For this study, a simulated corpus was generated so that the effect of each processing step can be checked against known ground truth. The corpus covers 36 months (January 2023 to December 2025) with between 266 and 382 postings per month and a dictionary of 30 skills. Each posting has a title (one of six IT roles), company, location, posting date, and free-text description. Skill prevalence follows author-specified trajectories: most skills drift slowly, while generative AI, large language models, MLOps, and Terraform grow rapidly from low starting levels. Skills co-occur through role-dependent affinities (for example, Kubernetes with DevOps and React with JavaScript). Skills are written using several surface forms, and some postings contain ambiguous phrases such as “Go the extra mile” or “Excel at communication”. Duplicate postings, empty descriptions, and invalid dates were injected to test the cleaning stage. Table I summarizes the corpus.

TABLE I CORPUS SUMMARY AND CLEANING OUTCOME

Item	Value
Postings generated (before injection of defects)	12,112
Postings after injecting duplicates	12,839
Removed: empty description	193
Removed: invalid posting date	100
Removed: duplicate postings	726
Unique postings analysed	11,820
Period / months	Jan 2023 – Dec 2025 / 36
Skills in dictionary	30
Mean skills per posting	3.67

B. Preprocessing

Records with empty descriptions or unparseable dates were removed. Duplicates were detected by hashing the combination of title, company, and whitespace-normalized lower-case description, and only the first occurrence was kept. Text cleaning preserves technical names such as C++, C#, .NET, and Node.js; punctuation is therefore not stripped before matching.

C. Skill Extraction and Normalization

Each skill has a canonical label and a list of aliases, for example “ML” and “machine learning” map to Machine Learning, “k8s” to Kubernetes, and “GenAI” to Generative AI. Aliases are matched with regular expressions that require non-alphanumeric boundaries, so that “Java” does not match inside “JavaScript”, “ML” does not match inside “MLOps”, and “C++” is matched as a whole token. Matching is case-sensitive for short or ambiguous names and case-insensitive for multi-word terms. Two context rules were added for ambiguous words: “Go” is not counted when followed by “the” or “beyond” or when part of “go-getter”, and “Excel” is not counted when followed by “at” or “in”. A baseline extractor without these context rules is used for comparison.

D. Demand and Emergence Measures

Each skill is counted at most once per unique posting. The demand share of skill s in month t is

$$\text{Share}(s, t) = 100 \times n(s, t) / N(t) \quad (1)$$

where $n(s, t)$ is the number of unique postings in month t mentioning s and $N(t)$ is the number of unique postings in that month. Co-occurrence between two skills is measured with the Jaccard index (the number of postings mentioning both divided by the number mentioning either) and with lift (observed joint frequency divided by the frequency expected if the skills were independent). A skill is called emerging when its mean share over the last six months is at least 3% and its ratio to the mean share over the first six months is high; this ranks skills by relative growth and avoids favouring skills that were already common.

E. Forecasting and Evaluation

The forecast target is $\text{Share}(s, t)$ for each skill. The first 27 months (2023-01 to 2025-03) are the initial training period and the last 9 months (2025-04 to 2025-12) are the test period. Forecasts are one month ahead and rolling: for each test month, models are fitted only on earlier months (expanding window), so training data always precedes test data. Five methods are compared: (1) Naive, which repeats the last observed value; (2) a three-month Moving Average; (3) Holt’s linear exponential smoothing with a damped trend ($\phi = 0.9$), with smoothing parameters selected per skill by grid search on in-sample one-step errors [9]; (4) Ridge regression and (5) a Random Forest with 150 trees, both trained on all skills pooled together using the three most recent values, the one-month and three-month differences, and the expanding mean as features, implemented with scikit-learn [6]. Accuracy is measured with Mean Absolute Error and Root Mean Squared Error, in percentage points of share, over all skills and test months:

$$\text{MAE} = (1/n) \sum | \hat{y}_i - y_i | \quad (2)$$

$$\text{RMSE} = \sqrt{ (1/n) \sum (\hat{y}_i - y_i)^2 } \quad (3)$$

where y_i is the observed share and $\hat{y}_i$ the forecast.

V. SYSTEM WORKFLOW

The workflow has six stages (Fig. 1). The entire pipeline was implemented in Python using pandas, NumPy, and scikit-learn, and every number in the next section is produced by running it end to end.

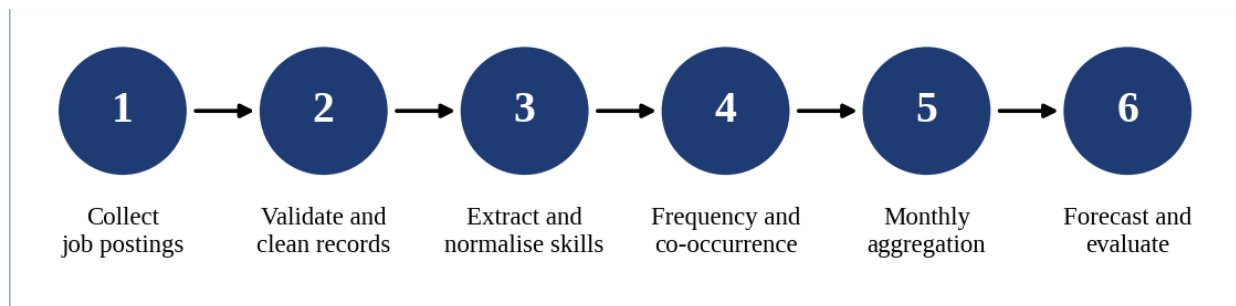


Fig. 1 Workflow of the proposed framework from raw postings to forecast evaluation.

VI. RESULTS AND DISCUSSION

All results below describe the simulated corpus. They show how the framework behaves and what its outputs look like; they should not be read as measurements of real hiring demand.

A. Effect of Cleaning and Extraction Quality

Cleaning removed 1,019 of 12,839 records (Table I), leaving 11,820 unique postings. Because extraction can be compared with the known skills of each simulated posting, precision, recall, and F1 are reported in Table II. Naive keyword matching found every true skill but produced 269 false matches from the ambiguous phrases; the context rules removed all of them. Perfect scores here are expected, because simulated descriptions are written from templates; on real text, recall would be lower and manual checking of a sample of matches remains necessary.

TABLE II SKILL EXTRACTION QUALITY ON THE SIMULATED CORPUS

Extractor	Precision	Recall	F1	False matches
Keyword matching (no context rules)	0.9938	1.0000	0.9969	269
Dictionary + context rules	1.0000	1.0000	1.0000	0

B. Skill Frequency and Co-occurrence

Table III lists the ten most frequently mentioned skills. Python (42.1%) and SQL (35.3%) lead, followed by Java, JavaScript, and AWS. A high share means a skill was mentioned in many postings; it does not show that the skill is the most valuable.

TABLE III TEN MOST FREQUENTLY MENTIONED SKILLS

Rank	Skill	Postings	Share (%)
1	Python	4,979	42.1
2	SQL	4,168	35.3
3	Java	3,042	25.7
4	JavaScript	2,866	24.2
5	AWS	2,854	24.1
6	Docker	1,875	15.9
7	React	1,863	15.8
8	Azure	1,725	14.6
9	Excel	1,658	14.0
10	Machine Learning	1,624	13.7

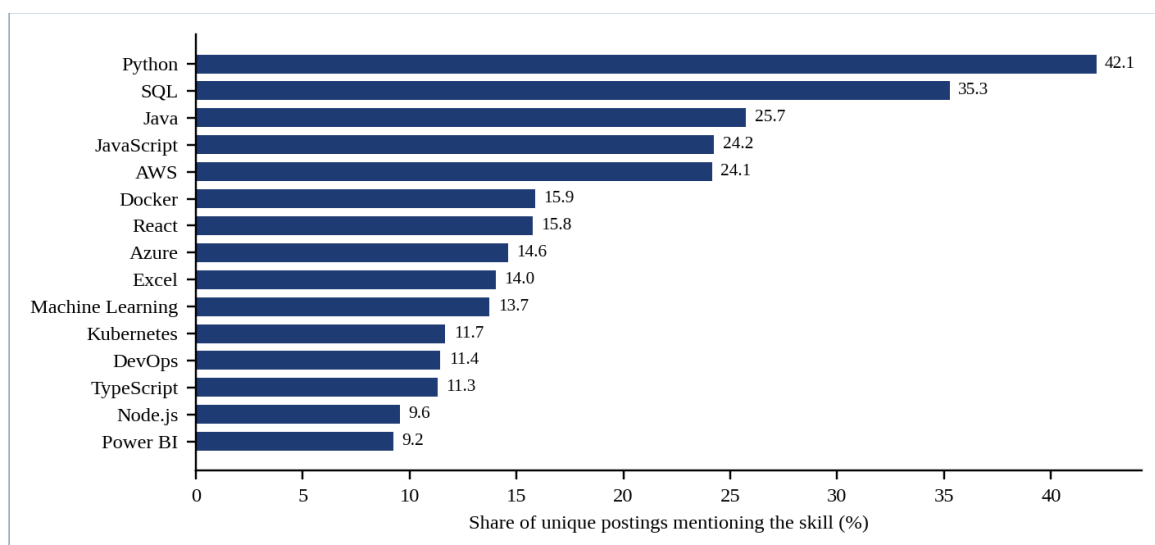


Fig. 2 Fifteen most frequently mentioned skills as a share of unique postings.

Co-occurrence analysis recovers the skill groupings built into the corpus (Table IV). The strongest pair by Jaccard index is JavaScript with React (0.30; lift 2.4), followed by Python with SQL and Kubernetes with DevOps. Python and SQL appear together often (1,188 postings) mainly because both are individually common, which is why their lift (1.74) is close to 1.

TABLE IV SKILL PAIRS WITH THE HIGHEST CO-OCCURRENCE

Skill A	Skill B	Postings	Jaccard	Lift
JavaScript	React	1,094	0.301	2.42
Python	SQL	1,890	0.260	1.08
Kubernetes	DevOps	514	0.232	3.25
JavaScript	TypeScript	761	0.221	2.35
Python	Machine Learning	1,188	0.219	1.74
Python	AWS	1,387	0.215	1.15
SQL	Java	1,197	0.199	1.12
TypeScript	React	529	0.198	2.51

C. Emerging Skills

Table V ranks skills by the ratio of their mean share in the last six months to that in the first six months, among skills reaching at least 3% share. LLM and Generative AI show the largest relative growth, rising from 0.7% and 1.3% to 10.7% and 13.9% respectively, followed by MLOps, Terraform, and TypeScript. Established skills such as Python and SQL change little in relative terms. Fig. 3 shows that the fastest-growing skills accelerate late in the period, which is the pattern built into the simulation.

TABLE V EMERGING SKILLS (MEAN SHARE, FIRST VS. LAST SIX MONTHS)

Skill	First 6 months (%)	Last 6 months (%)	Change (pp)	Ratio
LLM	0.7	10.7	+10.0	15.7×
Generative AI	1.3	13.9	+12.5	10.4×
MLOps	0.8	3.9	+3.1	4.7×
Terraform	4.3	9.9	+5.6	2.3×
TypeScript	7.0	14.8	+7.9	2.1×
PyTorch	4.2	8.0	+3.8	1.9×
Kubernetes	8.7	14.4	+5.7	1.7×
Go	3.4	5.5	+2.1	1.6×

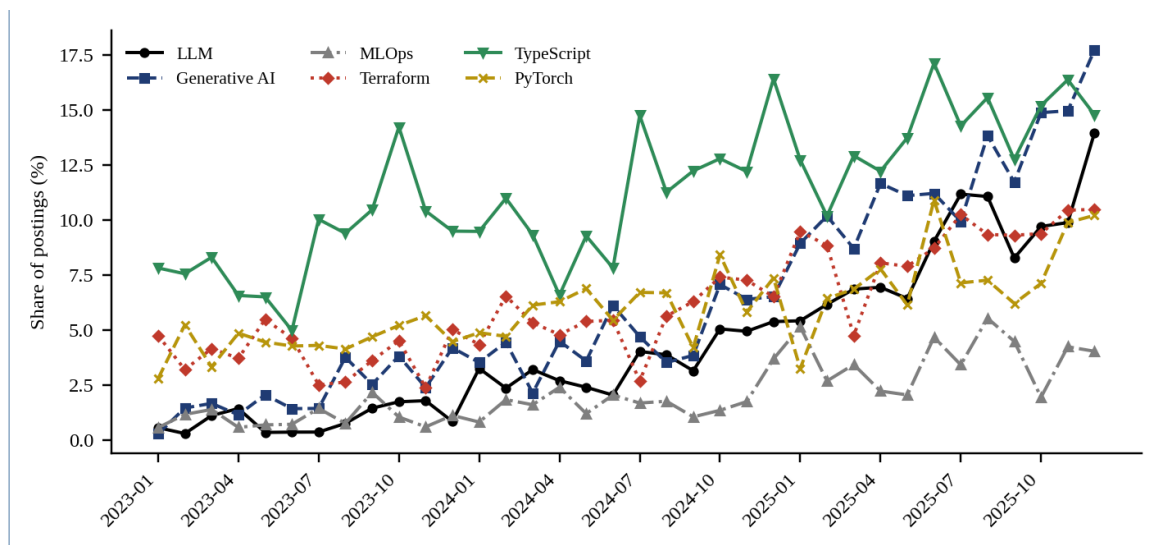


Fig. 3 Monthly share of the six fastest-growing skills.

D. Forecasting Results

Table VI reports rolling one-step-ahead accuracy over the nine test months for all 30 skills. The naive baseline has the highest error (MAE 1.81). Holt damped trend is best, with MAE 1.49 and RMSE 2.01, an improvement of 18.0% in MAE over the baseline. The moving average (13.3%) and Ridge regression (13.8%) are close behind, whereas the Random Forest improves on the baseline by only 7.4% and is worse than the moving average.

TABLE VI ONE-STEP-AHEAD FORECAST ACCURACY (PERCENTAGE POINTS)

Method	MAE	RMSE	MAE vs. naive
Naive	1.81	2.42	—
Moving average (3)	1.57	2.09	-13.3%
Holt damped trend	1.49	2.01	-18.0%
Ridge (lag features)	1.56	2.10	-13.8%
Random Forest (lag features)	1.68	2.20	-7.4%

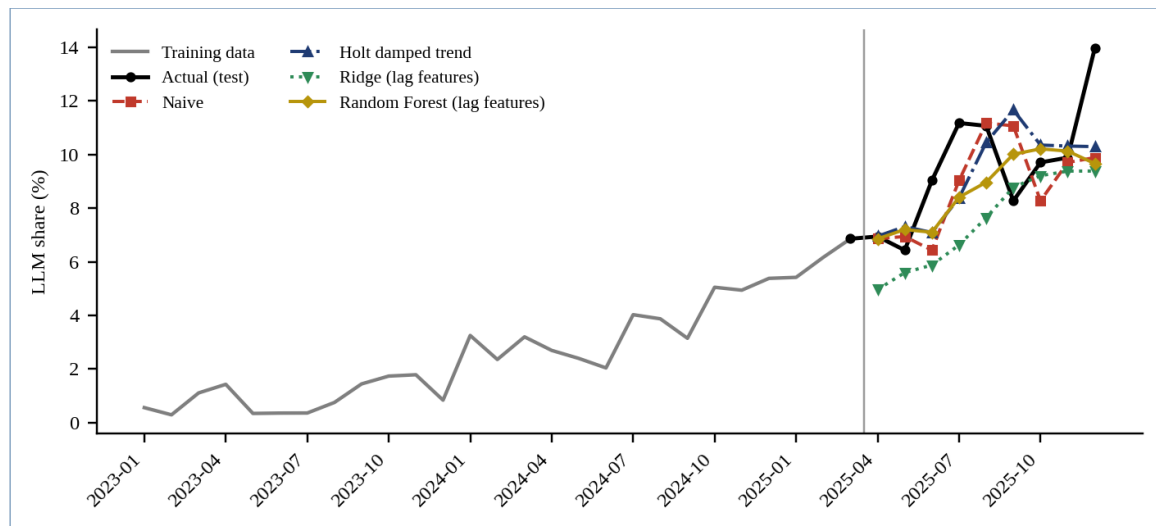


Fig. 4 Actual and forecast monthly share of LLM during the test period.

Results differ by skill. For the six emerging skills, the damped-trend model had the lowest MAE for 3 of 6 skills; the Naive method was best for 1, the moving average for 1, and the Random Forest for 1. Two points explain the overall pattern. First, a monthly share computed from roughly 300 postings carries sampling noise of about 1.2 to 2.4 percentage points for skills with 5% to 25% prevalence, so errors of about 1.5 points are comparable to the noise level and gains over the baseline are necessarily modest. Second, tree ensembles predict averages of values seen in training and so cannot extrapolate a rising series beyond its past range, which is visible for LLM in Fig. 4. Models with an explicit trend component are better suited to emerging skills.

The test period is only nine months and a single split was used, so the ranking of methods is indicative and no statistical test of differences was performed.

VII. LIMITATIONS AND ETHICAL CONSIDERATIONS

The most important limitation is that the corpus is simulated: skill prevalence, growth patterns, and co-occurrence were specified by the authors, and extraction is evaluated on template-based text. The results therefore validate the pipeline and illustrate its outputs, but make no claim about actual employer demand. On real data, vacancies may not represent all employers, regions, or hiring channels; advertisements can be duplicated, incomplete, expired, or written for search visibility; skill dictionaries can miss emerging terms or match ambiguous words; and forecasts can be unreliable when series are short, sparse, or affected by platform changes. Only data that can lawfully be accessed and reused should be analysed, and findings should be limited to the dataset studied.

VIII. CONCLUSION AND FUTURE WORK

This paper presented a reproducible framework for measuring IT skill demand in job postings, identifying emerging skills, and forecasting skill share with time-ordered evaluation. On a simulated corpus of 11,820 unique postings,

context-aware extraction eliminated the false matches of keyword matching, relative-growth ranking surfaced the fastest-growing skills, and damped-trend smoothing improved on the naive baseline by 18.0% in MAE, whereas a Random Forest did not improve on a moving average.

Future work will apply the same code to a documented public job-posting dataset, validate extraction by manual review of a sample, add location-wise comparison where reliable location data exist, evaluate forecasts over several rolling test windows and longer horizons, and add an interactive dashboard.

REFERENCES

- [1]. R. Boselli, M. Cesarini, and M. Mezzananza, "Classifying online job advertisements through machine learning," *Future Generation Computer Systems*, vol. 86, pp. 319–328, 2018, doi: 10.1016/j.future.2018.03.035.
- [2]. L. Samek, M. Squicciarini, and E. Cammeraat, "The human capital behind AI: Jobs and skills demand from online job postings," *OECD Science, Technology and Industry Policy Papers*, no. 120, OECD Publishing, 2021, doi: 10.1787/2e278150-en.
- [3]. S. Chaturvedi, K. Mahajan, and Z. Siddique, "Using domain-specific word embeddings to examine the demand for skills," IZA Discussion Paper No. 16593, 2023; also published in *Research in Labor Economics*, vol. 52B, pp. 171–223, 2024.
- [4]. E. Senger, M. Zhang, R. van der Goot, and B. Plank, "Deep learning-based computational job market analysis: A survey on skill extraction and classification from job postings," in *Proc. NLP4HR 2024*, pp. 1–15, doi: 10.18653/v1/2024.nlp4hr-1.1.
- [5]. Z. Ao, G. Horváth, C. Sheng, Y. Song, and Y. Sun, "Skill requirements in job advertisements: A comparison of skill-categorization methods based on wage regressions," *Information Processing & Management*, vol. 60, no. 2, art. 103185, 2023, doi: 10.1016/j.ipm.2022.103185.
- [6]. F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [7]. L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [8]. R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, 3rd ed. Melbourne, Australia: OTexts, 2021.
- [9]. E. S. Gardner Jr. and E. McKenzie, "Forecasting trends in time series," *Management Science*, vol. 31, no. 10, pp. 1237–1246, 1985.