Applying Statistical Methods to Analyze Software Engineering Skills: A Data-Driven Approach to Maximizing Internship Success

Tyler Mong[1]

[1]School of Business - Computer Science, Stockton University

Abstract

This paper analyzes 199 software engineering internship listings to identify the most in-demand technical skills. Python, C++, and Java were the most frequently mentioned skills, appearing in 38%, 24%, and 21% of listings, respectively. The data follows a right-skewed distribution, indicating that a small number of skills are dominant. Furthermore, these dominant skills are typically widely adopted and stable technologies, rather than newer, trendier ones. This analysis has several limitations, particularly the small sample size, and future improvements could include increasing the sample size and integrating a machine learning model for more advanced analysis.

Introduction

The software engineering job market continues to be in a highly competitive state, especially for those trying to break into tech by applying to new graduate and internship roles. Companies are increasingly seeking interns who possess strong technical abilities and can demonstrate their skills through live technical interviews. Additionally, with waves of layoffs and the rapid rise of artificial intelligence, the average computer science student is facing increased pressure to stand out in a crowded field. As such, it is essential for students to understand which skills are most in demand in order to remain competitive when applying to internship roles.

This study explores the skills required in software engineering internship positions by analyzing approximately 200 internship listings scraped from various job boards. The listings represent a wide range of companies, from big tech companies to smaller startups, providing a broad perspective on current industry expectations. The primary objective of this analysis is to identify the most in-demand technical skills across software engineering internships.

Through this research, statistical methods taught throughout the CSCI-3327 Probability and Applied Statistics course at Stockton University are applied to a series of sample problems, and the results are analyzed.

Methodologies

Data Collection

Internship listings were scraped using a Java-based script with the *jsoup*[1] library, which is built to parse webpage data. The data collection process was carried out as follows:

1. **Link Scraping**: Internship listing links were scraped from two GitHub repositories: Summer 2025 Tech Internships by Pitt CSC & Simplify, and Summer 2025 Tech Internships by Vansh & Ouckah. These repositories provided collections of links to internship listings hosted on various platforms.
2. **Duplicate Link Filtering**: Duplicate links were removed from the dataset to ensure each internship listing was only included once.
3. **Platform Link Filtering**: The remaining links were further filtered to include only those hosted on the Workday, Greenhouse, and Lever job platforms. Although the dataset spans 2,173 links across 436 different job boards, the three aforementioned platforms together account for approximately 28% of all links, despite representing only 0.69% of the total number of job boards.
4. **Data Scraping**: After filtering, the remaining internship listings were scraped for key information, including company name, job title, location, skills, and listing URL. Skills were extracted by identifying technical keywords[2], such as programming languages, frameworks, and tools,

[1] *jsoup*: the Java HTML parser, built for HTML editing, cleaning, scraping, and XSS safety. Available at: `https://jsoup.org/`.

[2] A full list of keywords can be found in: `src/main/java/dev/tylermong/jobanalyzer/util/SkillKeywords.java`

mentioned in the job description. Skills were standardized to be case-insensitive to group together the same skills with different capitalizations.

Statistical Analysis

The processed data was analyzed using a Java-based approach, where each skill keyword was stored as a key in a map, with the corresponding frequency of occurrences as the value. These key-value pairs were then written to a file for easy viewing. This simple frequency analysis lays the foundation for potential further statistical exploration, such as identifying trends or comparing skill frequencies across different internship listings.

Results

Skill Frequency Distribution

Skill	#	%
Python	75	38%
C++	47	24%
Java	41	21%
Algorithms	37	19%
AI	34	17%
Agile	26	13%
JavaScript	25	13%
Linux	22	11%
C	21	11%
SQL	19	10%

Table 1: *Top 10 most frequent skills across 199 internship listings.*

As shown in the Table 1, Python leads as the most popular skill, appearing in 38% of internship listings, followed by C++ and Java at 24% and 21% respectively. The median skill frequency occurrence is 1%, with the mean being 2.9%. This disparity indicates a heavily right-skewed distribution, where a small set of highly demanded skills dominates the market, and the majority of skills form a long tail of lower demand.

Problems and Solutions Using Scraped Data

1.1 Are some skills more common than others? Given below are the frequency of skills for 20 randomly selected software engineering skills:

2	6	4	7	18
1	17	3	75	34
10	37	16	5	21
9	25	8	41	14

Table 2: *20 randomly selected skill occurrences from 199 internship listings.*

a. Construct a relative frequency histogram for this data.

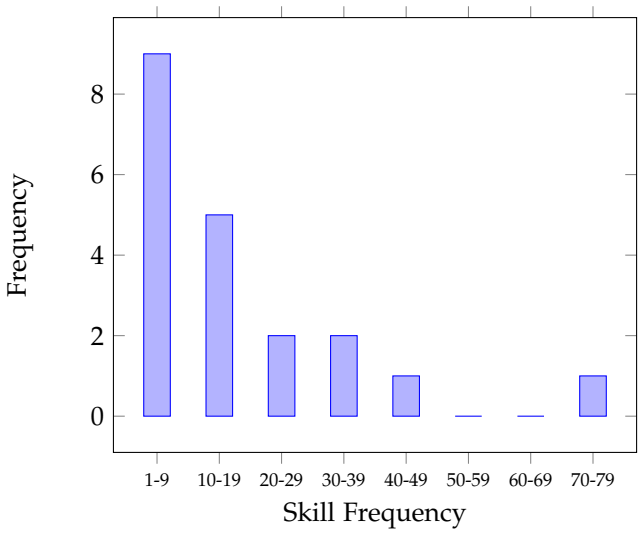

Figure 1: *Relative Frequency Histogram of Skill Occurrences*

2.1 A company is reviewing four internship applications with each applicant highlighting a different primary skill (Python, C++, Java, AI). The company random selects two applicants to interview.

a. List all possible pairs of applicants selected.

{(Python, C++), (Python, Java), (Python, AI), (C++, Java), (C++, AI), (Java, AI)}

b. Assign reasonable probabilities to each pair.

All pairs have an equal probability of $\frac{1}{6}$.

c. Find the probability that the applicant with Java as their primary skill is selected for an interview.

$$P(\text{Java selected}) = \frac{3}{6} = \frac{1}{2}$$

2.2 An intern needs to attend two different training sessions on separate days: one about programming languages (6 options) and one about software engineering methodologies (7 options). How many different two-session training combinations can the intern choose?

$$6 \times 7 = 42$$

2.3 An internship program is reviewing applicants for a software engineering position. There are 52 applicants, each with one primary skill: Python, C++, Java, or AI. The skills are evenly divided, meaning there are 13 applicants for each skill.

a. If the first two selected applicants both specialize in Python, what is the probability that the next three selected applicants will also specialize in Python?

$$\begin{aligned} P(\text{3 more Python selected}) &= \frac{11}{50} \times \frac{10}{49} \times \frac{9}{48} \\ &= 0.00841 = 0.84\% \end{aligned}$$

b. If the first three selected applicants all specialize in Python, what is the probability that the next two selected applicants will also specialize in Python?

$$\begin{aligned} P(\text{2 more Python selected}) &= \frac{10}{49} \times \frac{9}{48} \\ &= 0.03826 = 3.82\% \end{aligned}$$

c. If the first four selected applicants all specialize in Python, what is the probability that the next selected applicant will also specialize in Python?

$$\begin{aligned} P(\text{1 more Python selected}) &= \frac{9}{48} \\ &= 0.1875 = 18.75\% \end{aligned}$$

2.4 Two events, A (an applicant knowing Python) and B (an applicant knowing C++), are such that $P(A) = .2$, $P(B) = .3$, and $P(A \cup B) = .4$. Find the following:

a. $P(A \cap B)$

$$\begin{aligned} P(A \cap B) &= P(A) + P(B) - P(A \cup B) \\ P(A \cap B) &= 0.2 + 0.3 - 0.4 = 0.1 \end{aligned}$$

b. $P(\overline{A} \cup \overline{B})$

$$\begin{aligned} P(\overline{A} \cup \overline{B}) &= 1 - P(A \cap B) \\ P(\overline{A} \cup \overline{B}) &= 1 - 0.1 = 0.9 \end{aligned}$$

c. $P(\overline{A} \cap \overline{B})$

$$\begin{aligned} P(\overline{A} \cap \overline{B}) &= 1 - P(A \cup B) \\ P(\overline{A} \cap \overline{B}) &= 1 - 0.4 = 0.6 \end{aligned}$$

d. $P(\overline{A}|B)$

$$\begin{aligned} P(\overline{A}|B) &= \frac{P(\overline{A} \cap B)}{P(B)} \\ P(\overline{A} \cap B) &= P(B) - P(A \cap B) = 0.3 - 0.1 = 0.2 \\ P(\overline{A}|B) &= \frac{0.2}{0.3} = \frac{2}{3} \end{aligned}$$

2.5 Five interns are randomly selected from 52 total interns, who are divided evenly among 4 different teams, based on their preferred language (frontend, backend, data science, and AI/ML). What is the probability that all five selected new hires are from the same team?

$$\begin{aligned} \binom{52}{5} &= 2,598,960 \text{ total ways} \\ \binom{13}{5} &= 1287 \times 4 = 5148 \text{ favorable ways} \\ P(\text{All same team}) &= \frac{5148}{2,598,960} = 0.00198 = 0.198\% \end{aligned}$$

3.1 An application randomly maps three input field labels to three corresponding data fields. If the labels are assigned randomly, let Y be the number of labels correctly matched to their intended fields. Determine the probability distribution of Y.

Correct order: (1, 2, 3)

$1,2,3$: 3 correct
$1,3,2$: 1 correct
$2,1,3$: 1 correct
$2,3,1$: 0 correct
$3,1,2$: 0 correct
$3,2,1$: 1 correct

$$\begin{aligned} P(Y = 0) &= \frac{2}{6} = \frac{1}{3} \\ P(Y = 1) &= \frac{3}{6} = \frac{1}{2} \\ P(Y = 2) &= 0 \\ P(Y = 3) &= \frac{1}{6} \end{aligned}$$

3.2 A fault-tolerant distributed software system includes four redundant service modules that perform the same task. Each module has a 20% chance of crashing within 1000 hours. The system is designed to remain operational as long as at least two of the four modules are running. Assume each module fails independently.

a. What is the probability that exactly two out of the four modules survive past 1000 hours?

$$\begin{aligned} P(Y=2) &= \binom{4}{2} \cdot (0.8)^2 \cdot (0.2)^2 \\ &= 6 \cdot 0.64 \cdot 0.04 = 0.1536 \end{aligned}$$

b. What is the probability that the system as a whole continues functioning beyond 1000 hours?

$$\begin{aligned} P(\text{system works}) &= P(Y \geq 2) \\ P(Y \geq 2) &= 1 - P(Y=0) - P(Y=1) \\ P(Y=0) &= \binom{4}{0} \cdot (0.8)^0 \cdot (0.2)^4 = 0.0016 \\ P(Y=1) &= \binom{4}{1} \cdot (0.8)^1 \cdot (0.2)^3 = 0.0256 \\ P(Y \geq 2) &= 1 - 0.0016 - 0.0256 = 0.9728 \end{aligned}$$

3.3 Suppose that 30% of software engineering internship applicants report proficiency in Python. Applicants are interviewed one at a time and selected at random from the pool. What is the probability that the first applicant with Python experience is found on the fifth interview?

$$\begin{aligned} P(Y=5) &= q^{y-1} \cdot p \\ &= (0.70)^4 \cdot 0.30 \\ &= 0.07203 \end{aligned}$$

3.4 10% of applicants in an applicant pool lack essential SQL skills. Applicants are interviewed one by one and randomly selected. What is the probability that the first candidate with strong SQL skills will be found in the second interview?

$$\begin{aligned} P(Y=2) &= (0.1)^1 \cdot (0.9) \\ &= 0.09 \end{aligned}$$

3.5 An applicant pool contains 10 applicants, of which 5 list Python as a skill, 2 list C++, and 3 list Java. If 3 applicants are selected one at a time without replacement, what is the probability that all three selected applicants list Python on their resume?

$$\begin{aligned} P(Y=5) &= \frac{\binom{5}{3}\binom{5}{0}}{\binom{10}{5}} \\ &= 0.0833 \end{aligned}$$

3.6 The number of bugs found in a code review follows a Poisson distribution with an average of 4 bugs per review. If more than 4 bugs are found in a review, the author must revise the entire code submission. What is the probability that a randomly selected review does not require a full revision?

$$P(Y \leq 4) = P(Y=0) + \ldots + P(Y=4)$$

$$\begin{aligned} P(Y=0) &= \frac{e^{-4} * 4^0}{0!} = 0.0183 \\ P(Y=1) &= \frac{e^{-4} * 4^1}{1!} = 0.0733 \\ P(Y=2) &= \frac{e^{-4} * 4^2}{2!} = 0.1465 \\ P(Y=3) &= \frac{e^{-4} * 4^3}{3!} = 0.1954 \\ P(Y=4) &= \frac{e^{-4} * 4^4}{4!} = 0.1954 \end{aligned}$$

$$0.0183 + 0.0733 + 0.1465 + 0.1954 + 0.1954 = 0.6289$$

3.7 A company processes large volumes of resumes using an automated parser. The average file size is 0.5 MB with a standard deviation of 0.01 MB. Using Tchebysheff's Theorem, find a lower bound for the number of resumes out of 400 total expected to have a file size between 0.48 MB and 0.52 MB.

$$\begin{aligned} k &= \frac{0.02}{0.01} = 2 \\ 1 &- \frac{1}{2^2} = 0.75 \\ 400 &\times 0.75 = 300 \end{aligned}$$

3.8 In a technical interview screening process, 20 interviewees are asked to select a coding challenge from a collection of 10 challenges. Eight choose either 4, 5, or 6. If the interviewees make their choices independently and each challenge is equally likely to be selected, what is the probability that 8 or more will select challenges 4, 5, or 6?

$$P(4) = P(5) = P(6) = \frac{1}{10}$$
$$P(4 \text{ or } 5 \text{ or } 6) = \frac{3}{10}$$
$$P(Y \geq 8) = 1 - P(Y \leq 7)$$

$$P(Y \leq y) = \sum_{y=0}^{y} \binom{n}{y} \cdot p^y \cdot q^{n-y}$$
$$P(Y \leq 7) = \sum_{y=0}^{7} \binom{20}{y} \cdot (0.3)^y \cdot (0.7)^{20-y}$$
$$= 0.7723$$
$$P(Y \geq 8) = 1 - 0.7723$$
$$= 0.2277$$

4.1 A software company manages a server with a capacity of 150 TB that is allocated at the beginning of each week. The weekly data usage shows a pattern that increases steadily up to 100 TB, then levels off between 100 TB and 150 TB. If Y denotes data usage in tens of terabytes, the relative frequency of data usage can be modeled by:

$$f(y) = \begin{cases} y, & 0 \leq y \leq 1 \\ 1, & 1 < y \leq 1.5 \\ 0, & \text{elsewhere} \end{cases}$$

a. Find $F(y)$.

$$y < 0 : F(y) = \int_{-\infty}^{y} 0dy$$
$$= 0$$
$$0 \leq y \leq 1 : F(y) = \int_{-\infty}^{0} 0dy + \int_{0}^{y} ydy$$
$$= 0 + \left[\frac{y^2}{2}\Big|_0^y\right]$$
$$= \frac{y^2}{2}$$
$$1 \leq y \leq 1.5 : F(y) = \int_{-\infty}^{0} 0dy + \int_{0}^{1} ydy + \int_{1}^{y} 1dy$$
$$= 0 + \frac{1}{2} + \left[y\Big|_1^y\right]$$
$$= \frac{1}{2} + (y - 1)$$
$$= y - \frac{1}{2}$$
$$y > 1.5 : F(y) = \int_{-\infty}^{0} 0dy + \int_{0}^{1} ydy + \int_{1}^{1.5} 1dy + \int_{1.5}^{\infty} 0dy$$
$$= 0 + \frac{1}{2} + \frac{1}{2} + 0$$
$$= 1$$
$$F(y) = \begin{cases} 0, & y \leq 0 \\ \frac{y^2}{2}, & 0 \leq y \leq 1 \\ y - \frac{1}{2}, & 1 < y \leq 1.5 \\ 1, & y > 1.5 \end{cases}$$

b. Find $P(0 \leq Y \leq .5)$.

$$= F(0.5) - F(0)$$
$$= \frac{(0.5)^2}{2} - 0$$
$$= 0.125$$

c. Find $P(.5 \leq Y \leq 1.2)$.

$$= F(1.2) - F(0.5)$$
$$= \left[(1.2) - \frac{1}{2}\right] - \frac{(0.5)^2}{2} \quad = 0.575$$

4.2 The response time Y represents the time in milliseconds at which a system triggers an alert, has a probability density function given by

$$f(y) = \begin{cases} \frac{1}{2}, & 59 \leq y \leq 61 \\ 0, & \text{elsewhere} \end{cases}$$

Find $E(Y)$ and $V(Y)$.

$$E(Y) = \int_{\infty}^{\infty} yf(y)dy$$
$$= \int_{59}^{61} y \cdot \frac{1}{2}dy$$
$$= \frac{1}{2}\int_{59}^{61} ydy$$
$$= \frac{1}{2}\left[\frac{y^2}{2}\Big|_{59}^{61}\right]$$
$$= 60$$

$$
\begin{aligned}
V(Y) &= E(Y^2) - (E(Y))^2 \\
E(Y)^2 &= (60)^2 \\
&= 3600 \\
E(Y^2) &= \frac{1}{2}\int_{59}^{61} \left[\frac{y^3}{3}\Big|_{59}^{61}\right] \\
&= \frac{10,801}{3} \\
V(Y) &= \frac{10,801}{3} - 3600 \\
&= \frac{1}{3}
\end{aligned}
$$

4.3 A request was made to a server at random within a one-minute interval. The server was busy for the first 15 seconds of the minute. What is the probability that the request was made when the server was not busy?

$$
\begin{aligned}
P(\text{not busy}) &= \frac{T_{\text{not busy}}}{T_{total}} \\
&= \frac{45}{60} \\
&= 0.75
\end{aligned}
$$

4.4 A server processes a specific type of bulk data. The amount of data processed in one day can be modeled by an exponential distribution with $\beta = 4$ (measured in terabytes). What is the probability that the server will process more than 4 terabytes on a given day?

$$
\begin{aligned}
P(Y > y) &= e^{-\lambda y} \\
&= e^{-.25 \cdot 4} \\
&= 0.3678
\end{aligned}
$$

5.1 Two tickets are randomly assigned to one or more of three teams, A, B, and C. Let X denote the number of tickets assigned to team A and Y the number of tickets assigned to team B. Recall that each team can receive 0, 1, or 2 tickets.

a. Find the joint probability function for X and Y.

$$X, Y : \frac{\binom{2}{x} \cdot \binom{2-x}{y}}{3^2}$$

$0,0 : \frac{\binom{2}{0} \cdot \binom{2}{0}}{3^2} = \frac{1}{9}$ $\quad$ $0,1 : \frac{\binom{2}{0} \cdot \binom{2}{1}}{3^2} = \frac{2}{9}$ $\quad$ $0,2 : \frac{\binom{2}{0} \cdot \binom{2}{2}}{3^2} = \frac{1}{9}$

$1,0 : \frac{\binom{2}{1} \cdot \binom{1}{0}}{3^2} = \frac{2}{9}$ $\quad$ $1,1 : \frac{\binom{2}{1} \cdot \binom{1}{1}}{3^2} = \frac{2}{9}$ $\quad$ $1,2 : \text{N/A}$

$2,0 : \frac{\binom{2}{2} \cdot \binom{0}{0}}{3^2} = \frac{1}{9}$ $\quad$ $2,1 : \text{N/A}$ $\quad$ $2,2 : \text{N/A}$

	$Y(A)$		
$X(A)$	0	1	2
0	$\frac{1}{9}$	$\frac{2}{9}$	$\frac{1}{9}$
1	$\frac{2}{9}$	$\frac{2}{9}$	0
2	$\frac{1}{9}$	0	0

b. Find $F(1,0)$.

$$F(1,0) = \frac{2}{9}$$

5.2 Let X and Y be the input size and processing time of an algorithm, respectively, and have the joint probability density function given by

$$
f(x,y) = \begin{cases} k(1-y), & 0 \le x \le y \le 1, \\ 0, & \text{elsewhere} \end{cases}
$$

a. Find the value of k that makes this a probability density function.

1. $x : 0, y$
   $y : 0, 1$
2. $\int_{y=0}^{1}\int_{x=0}^{y} k(1-y)dxdy = 1$

3a. $\int_{x=0}^{y} k(1-y)dx$
$= ky(1-y)$

3b. $\int_{y=0}^{1} ky(1-y)dy$
$= \frac{k}{6}$

4. $\frac{k}{6} = 1$
   $k = 6$

b. Find $P(X \leq \frac{3}{4}, Y \geq \frac{1}{2})$.

1. $X \leq \frac{3}{4} : \left(0,\ \frac{3}{4}\right)$
   $Y \geq \frac{1}{2} : \left(\frac{1}{2},\ 1\right)$
   $Y\left(\frac{1}{2},\ \frac{3}{4}\right) : X\,(0,\ y)$
   $Y\left(\frac{3}{4},\ 1\right) : X\left(0,\ \frac{3}{4}\right)$

2. $\int_{y=1/2}^{3/4}\int_{x=0}^{y} 6(1-y)dxdy$
   $+ \int_{y=3/4}^{1}\int_{x=0}^{3/4} 6(1-y)dxdy$

$3a_1$. $\int_{x=0}^{y} 6(1-y)dx$
$= 6y - 6y^2$
$3a_2$. $\int_{x=0}^{3/4} 6(1-y)dx$
$= \frac{9}{2} - \frac{9}{2}y$

$3b_1$. $\int_{y=1/2}^{3/4} 6y - 6y^2 dy$
$= \frac{11}{32}$
$3b_1$. $\int_{y=3/4}^{1} \frac{9}{2} - \frac{9}{2}y dy$
$= \frac{9}{64}$

4. $\frac{11}{32} + \frac{9}{64}$
$= \frac{31}{64}$

5.3 The following,

$$f(x,y) = \begin{cases} 4xy, & 0 \leq x \leq 1, 0 \leq y \leq 1, \\ 0, & \text{elsewhere} \end{cases}$$

is a valid joint probability density function, representing the likelihood of encountering a bug based on the amount of recently written code (x) and the complexity of that code (y). Find the marginal density functions for X and Y.

$$\int_0^1 4xydy = 4x\left[\frac{y^2}{2}\Big|_0^1\right] = 2x$$
$$\int_0^1 4xydx = 4y\left[\frac{x^2}{2}\Big|_0^1\right] = 2y$$

5.4 In Exercise 5.2 a., we determined that

$$f(x,y) = \begin{cases} 6(1-y), & 0 \leq x \leq y \leq 1 \\ 0, & \text{elsewhere} \end{cases}$$

is a valid joint probability density function, with X and Y representing the input size and processing time of an algorithm, respectively. Are X and Y independent?

$$f_Y(y) = \int_{x=0}^{y} 6(1-y)dx \quad = 6y(1-y)$$
$$f_X(x) = \int_{y=x}^{1} 6(1-y)dy \quad = 3x^2 - 6x + 3$$

$$f(0.25, 0.5) = 6(1-0.5) = 3$$
$$f_X(0.25) = 3(0.25)^2 - 6(0.25) + 3 = 1.6875$$
$$f_Y(0.5) = 6(0.5)(0.5) = 1.5$$

$(1.6875)(1.5) = 2.53125 \neq 3 \quad \Rightarrow$ No, dependent.

Discussion

The analysis of software engineering internship postings provided relevant data regarding the current skills desired by employers. Python occurred the most frequently, appearing in 38% of listings. This is a significant jump from the second most popular skill, C++, which appeared in 24% of listings. Its rise in popularity is likely due to Python excelling in many areas, such as data science, machine learning, and web development, paired with its low learning curve.

The right-skewed distribution also provides important insight into skills, highlighting that there is a small set of highly demanded skills, such as those provided in Table 1, which dominate the market. While some technologies may be popular in developer circles due to their cutting-edge features, it is clear that the job market prefers technologies with wide adoption and long-term stability.

While this analysis provides a good starting point in uncovering broader skill trends, there are several limitations to the data that should be addressed.

To start, the 199 listings that the data is extracted from are a relatively small sample size when compared to the number of active internship and job listings. There are two main methods to remedy this. First, adding support for platforms other than Workday, Greenhouse, and Lever, and second, adding support for additional link scraping, so as not to be limited by the current two repositories.

Next, extending the analysis to include other data points, such as location, company sector, and company size, could allow for additional and more accurate insights. Along with this, exploring how trends shift over time would also be valuable.

Finally, with this additional data, adding a machine learning model to discover deeper patterns, such as skill clustering (e.g., Python, data science, and machine learning appearing together) or projecting current trends into the future, would add a further layer of insight into the data. A machine learning model could also be implemented in order to scrape skills and other keywords more effectively. Currently the description of each listing is scanned word-by-word, looking for matches against the VALID_SKILLS, from the SkillKeywords class. While the list is extensive and the scraping is relatively effective, the methodology is primitive and requires manual updating to add new skills.

Conclusion

This analysis highlights the most in-demand technical skills found in software engineering internship listings, with Python, C++, and Java at the forefront. The data suggests that companies widely favor widely adopted, stable, and long-standing technologies over newer, trend-driven ones. While the findings offer useful insights, they are limited by several factors, such as a limited sample size.