Job Posting Data Mining and Descriptive Analysis Report

Based on Multi-Platform Big Data (Zhilian, Boss, 51job)

Team Presentation of czw, wyg, zzh

November 27, 2025

Presentation Outline

1.1 Macro-Economic Context

- **Structural Adjustments:** The labor market is undergoing a significant shift from traditional manufacturing to high-tech and service industries.
- **Talent Mismatch:** There is a growing "skills gap" between university output and corporate requirements, creating friction in the employment market.
- **Digital Transformation:** Recruitment has moved almost entirely online, generating massive datasets that serve as a barometer for economic health.

1.2 The Value of Recruitment Data

For Job Seekers:

- Accurate salary benchmarking.
- Identification of high-demand skills (e.g., Python, AI).
- Geographic hotspot analysis.

For Researchers & Policy Makers:

- Real-time monitoring of industry trends.
- Analyzing the impact of policies on specific sectors (e.g., EduTech, Real Estate).

1.3 Technical Challenges in Data Mining

Why is this difficult?

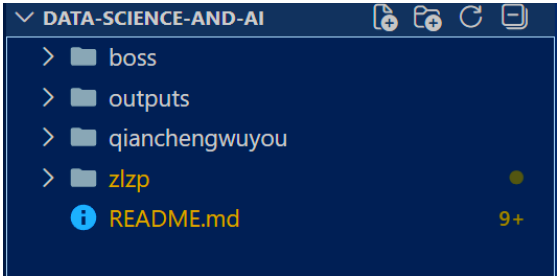
1. **Information Silos:** Data is fragmented across competing platforms with different data structures (JSON vs HTML).
2. **Anti-Scraping Mechanisms:**
   - **Network Level:** IP bans, frequency limits, WAF (Web Application Firewall).
   - **Application Level:** Custom font obfuscation (mapping numbers to icons), dynamic AJAX rendering, and complex Captchas (slide, click).
3. **Data Hygiene:** Issues with duplicate posts, vague salary ranges (e.g., "10k-50k"), and non-standardized job titles.

2.1 Team Division

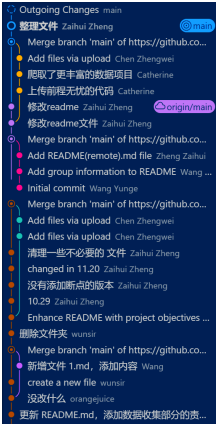
Member	Platform	Core Technical Contribution
CZW	Zhilian	Selenium Automation, **Incremental Write**
ZZH	Boss	API Reverse Engineering, **Resumable System**
WYG	51job	Stealth Injection, **Hybrid Parsing (JSON/DOM)**

2.2 Work Devision

workspace



commit history



3.1 Zhilian Zhaopin (Zhaopin.com)

Overview: One of China's earliest recruitment platforms, focusing on traditional white-collar and enterprise recruitment.

- **Data Characteristics:**
  - Highly structured standardized fields (Education, Experience).
  - Often displays salary in "Yearly" terms for senior roles.
- **Technical Architecture:**
  - Heavily reliant on server-side rendering mixed with dynamic AJAX for lists.
  - Anti-bot measures focus on login walls and frequency capping.

3.2 Boss Zhipin (Boss Direct Chat)

Overview: A mobile-first platform pioneering the "Direct Chat" model, dominating the tech and SME startup sectors.

- **Data Characteristics:**
  - Extremely high update frequency; jobs expire quickly.
  - Descriptions often contain colloquial language and emoji, requiring NLP cleaning.
- **Technical Architecture:**
  - **Most Difficult Target:** Uses rigorous Cookie encryption (seed/token) and parameter signing.
  - Data is often loaded via encrypted JSON APIs rather than static HTML.

3.3 51job (Qiancheng Wuyou)

Overview: A comprehensive HR service provider covering all industries, from blue-collar to executive search.

- **Data Characteristics:**
  - Massive volume of listings.
  - Legacy data structures mixed with new React-based frontends.
- **Technical Architecture:**
  - Recently updated to a more modern frontend.
  - Embeds structured data within ‘sensorsdata‘ attributes for analytics, which we exploit for scraping.
  - Uses ‘webdriver‘ detection (checking for automation flags).

4.1 Zhilian Zhaopin Strategy (CZW)

Core Tech: Selenium Automation + Dynamic Rendering

- **Mechanism:**
  - Uses Selenium to drive Chrome, simulating real user visits.
  - Controls pagination via URL parameters (`p=PageNum`).
- **Anti-Scraping Strategy:**
  - Uses `WebDriverWait` to handle dynamic page rendering.
  - Implements `random.sleep(3, 7)` to simulate human operation intervals and reduce blocking risks.
- **Data Storage (Incremental):**
  - Supports real-time writing to Excel/CSV. Data is saved immediately after each page is crawled to prevent loss during crashes.

4.2 Boss Zhipin Strategy (ZZH)

Core Tech: Requests Reverse API + Resumable + Dual-Stage

- **Mechanism (High Efficiency):**
  - Directly requests the backend JSON API (`/search/joblist.json`) instead of rendering pages.
- **Dual-Stage Collection:**
  - **List Fetch:** Batch retrieval of Job IDs and basic info.
  - **Detail Fetch:** HTML parsing of description based on IDs.
- **Key Features:**
  - **Resumable:** Records progress locally; resumes from the last break-point after interruption.
  - **Risk Control:** Detects error codes (e.g., Code 37) to prompt Cookie updates.
  - **Cleaning:** Built-in Regex to filter specifically for "Finance/Bank/Insurance" roles.

4.3 51Job Strategy (WYG)

Core Tech: Selenium + Stealth + Hybrid Parsing + DB

- **Anti-Scraping Evolution:**
  - Integrates `selenium-stealth` or injects `stealth.min.js` to hide WebDriver fingerprints.
- **Hybrid Parsing Strategy:**
  - **Priority:** Extracts structured JSON directly from the DOM's `sensorsdata` attribute (High Accuracy).
  - **Fallback:** Reverts to CSS selectors if JSON extraction fails.
- **Interaction & Storage:**
  - Handles "Infinite Scroll/Lazy Load" via script control.
  - Includes logic for Salary Unit Cleaning and **MongoDB** insertion.

4.3 51Job: Hybrid Parsing Code

```
# From 51job_spider.py
try:
    # Priority: Parse JSON from sensorsdata
    sensors = job_elem.get_attribute('sensorsdata')
    info = json.loads(_html.unescape(sensors))
    job_salary = info.get('jobSalary', '')
except:
    # Fallback: CSS Selector
    job_salary = safe_text(it, ['.sal', '.salary'])

```

5.1 Data Fusion Advanced Cleaning

- **Integration:** Merge data from MongoDB (51job), Excel (Zhilian), and Excel (Boss).
- **Standardization:**
  - Map diverse columns to a unified schema (Title, Salary, Company, Loc).
  - Normalize salary ranges to "Monthly Average" (e.g., convert "Yearly" to "Monthly").

5.2 Descriptive Analysis Plan

1. **Economic Insights**
   - Salary Heatmaps by City.
   - Industry premiums (Finance vs. Tech).

2. **Skill Mining**
   - Apply NLP (Jieba) on the captured "Job Descriptions".
   - Generate word clouds for top required skills.

Conclusion

Summary

- **Zhilian:** Incremental automation ensures data safety.
- **Boss:** API reverse engineering maximizes efficiency and handles risks.
- **51Job:** Stealth technology and hybrid parsing conquer complex anti-scraping.