Employees' Education Information Scraping and Analysis Based on AI

LI Xianda xliep@connect.ust.hk
WU Jiaquan jwubn@connect.ust.hk
YAN Ke kyanab@connect.ust.hk
YU Yipeng yyubo@connect.ust.hk
ZHAO Siheng szhaoam@connect.ust.hk

Abstract

Aimed at doing a comprehensive and effective analysis for the future prospects of artificial intelligent companies, we try to start from the intelligence resources of the companies, especially their employees' education background and professional qualification. We firstly collect and integrate relevant data from LinkedIn, establish a reliable scoring system, and summarize data to a company level for analysis and data visualization. For other issues encountered in data extraction, such as pdf resume parsing problem, we use some feasible methodologies to complete the analysis. In the end, we obtain an effective and reliable automated framework for further analysis use.

1 Introduction

In this paper, first, we will learn about the latest articles on neural networks for building data classification and get relevant inspiration or technical applications. Then, we tried to use the web data scraping to download the resume data we need from the LinkedIn and other job search websites. Next, we establish an automatic scoring system through Excel Fuzzy Lookup and VBA to rate each employee from data we obtained and then visually analyze the data. In this case, when encountering a resume on the website in pdf or when we only have a pdf format resume, we try to use pdf parsing or natural language processing to parse and extract the content, successfully crawling our data.
In order to enhance the viability of our project, we will use the following methodologies and will give a brief introduction to them.
Neural networks: mathematical models or computational models that mimic the structure and function of biological neural networks for estimating or approximating functions in the field of machine learning and cognitive science. Modern neural networks are a kind of nonlinear statistical data modeling. The neural network is usually optimized by a learning method based on mathematical statistics.

Web scraping: the process of automatically mining data or collecting information from the World Wide Web. It is a field with active developments sharing a common goal with the semantic web vision, an ambitious initiative that still requires breakthroughs in text processing, semantic understanding, artificial intelligence and human-computer interactions. Current web scraping solutions range from the ad-hoc, requiring human effort, to fully automated systems that are able to convert entire web sites into structured information, with limitations. Web scraping a web page involves fetching it and extracting from it. Fetching is the downloading of a page. Therefore, web crawling is a main component of web scraping, to fetch pages for later processing. Once fetched, then extraction can take place. The content of a page may be parsed, searched, reformatted, its data copied into a spreadsheet, and so on. Web scrapers typically take something out of a page, to make use of it for another purpose somewhere else. An example would be to find and copy names and phone numbers, or companies and their URLs, to a list (contact scraping).
PDFMiner: a tool for extracting information from PDF documents. Unlike other PDF-related tools, it focuses entirely on getting and analyzing text data. PDFMiner allows one to obtain the exact location of text in a page, as well as other information such as fonts or lines. It includes a PDF converter that can transform PDF files into other text formats (such as HTML).

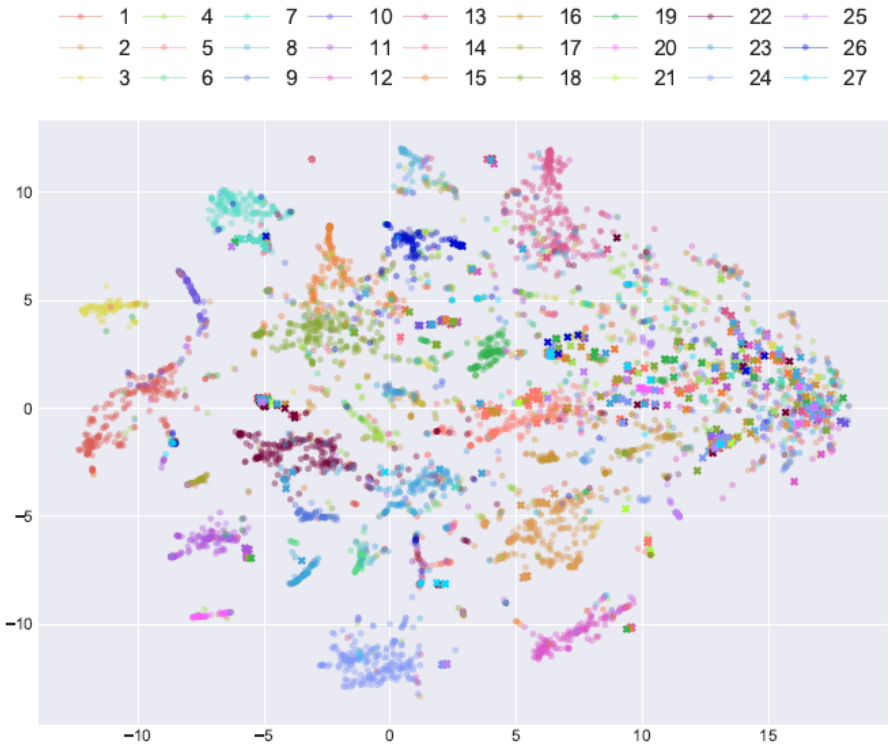
2 Related-work

In the recruitment practice of HR, “person-post matching” is the essence of the whole, and it is necessary to “get the person” and “the person fits the job”. This is a double matching process with some complexity. As the first "gatekeeper" in the process of matching people, the screening and review of resumes is the primary and very important work of HR.
Searching in a large number of resumes is the first step in HR screening resumes. Whether it's headhunting or corporate HR, it takes a lot of time every day to search for excellent resumes and job search information, not to mention 70% of the energy of domestic small and medium-sized companies HR will be spent on resume search and review. Suppose a HR looks at a resume every 30 seconds, and 200 resumes take about 1.7 hours. If you enter the interview at a 10% rate, the remaining 180 resumes that have spent 1.5 hours reading HR will eventually lie in the trash. With this accumulation, HR will throw away 45,000 resumes a year. This is equivalent to one HR and one month of working time. In other words, HR will be "wasted" one month in a year. But unfortunately, the search results of resumes are often inaccurate and even ridiculous.
Therefore, building a system that automatically identifies resumes and recommends them to the right positions is essential to optimizing the recruitment process. Based on this idea, and combining the two methods practiced in the AI course project: data scarping from LinkedIn and automated scoring system construction, we learned an article to solve the above problem.

The name of the article is called 'Domain Adaptation for Resume Classification Using Convolutional Neural Networks[1]', which is published in AIST proceedings: Springer's Lecture Notes in Computer Science (LNCS) series.

In this article, they have devised a resume classification method which is able to exploit the information contained in vast amounts labelled job description data in order to achieve higher accuracy. Since resumes are more sensitive data and difficult to obtain, compared to job summaries, we trained the proposed model only on job summaries and tested its performance on resume data with the same job category labels. A convolutional neural network for short text classification using word embeddings was trained and validated on 85,000 short job summaries mined from Indeed. Then this network was used to classify a set of 523 candidate resumes and compared with a simple but effective fastText model. Our method achieved 74.88% accuracy on job classification task and 40.15 % on resume classification, thereby outperforming the existing fastText model by more than 6% on resume classification task and 3% on the job description task. Moreover, we applied our method to a small imbalanced dataset consisting of 98 children dream job descriptions. In this task CNN outperformed fastText by 22%.

Given the fact that no labels were used from resume data for training or validation, we consider CNN for short classification to be useful in a domain adaptation scenario. An interesting direction for future work would be to study whether the results can be improved by leveraging a small number of labelled resume samples to fine-tune the CNN model.



[1] https://arxiv.org/abs/1707.05576

Figure 1: t-SNE visualization using the CNN first layer outputs on job and resume data

3 Data scarping from LinkedIn

In the AI and FinTech companies, the talent capital plays a critical part in their steady growth. We expect that the great companies should be led by top-tier researchers and scientists in this field. LinkedIn, as a worldwide adopted online social network of professionals, gives us an opportunity to extract employers' education background in a company and make evaluation of the employers.

We searched for several methods online and made a plan to conduct the data scraping based on Python3. The packages we used include *.utils, .exceptions, time,* and *lxml*. A good data scraping is very helpful for scoring system in the later stage.

Table 1: Demonstration of data scraping codes

```
class LinkedinItem(object):

    attributes_key = ['volunteerings', 'last_name',
'number_recommendations',
                      'number_connections',
'current_location', 'honors', 'first_name',
                      'current_title', 'test_scores',
'current_industry', 'languages',
                      'similar_profiles', 'interests',
'has_profile_picture', 'current_education',
                      'educations', 'experiences',
'groups', 'organizations', 'certifications',
                      'name', 'skills', 'websites',
'summary', 'project', 'courses',
                      'publications', 'recommendations']

    def __init__(self, url=None, html_string=None,
crequest=None):
        # if you want put the html text directly
        self.url = url
        self.html_string = html_string
        if self.html_string is not None:
            self.tree = html.fromstring(self.html_string)
```

We tried several times. However, our LinkedIn accounts were banned due to data scraping every time. We can only get limited information in every trial.

To make sure enough data for scoring system, we finally decided to scrap data from LinkedIn by manual operation. Due to networking system of LinkedIn, we cannot see all the staff of the target companies. In average, we scrap 20-40 employers for each

company. We searched for the Bachelor. Master, Phd background of the staff and the relevant universities. Totally, we got 100+ employers' data.

Company	Name	Bachelor	Institution Name
Qxbranch	Kyle Garman	Northwestern University	NORTHWESTERN UNIVERSITY
Qxbranch	Jinesh Patel	University of Exeter	UNIVERSITY OF EXETER
Qxbranch	Jack L.	North Carolina State University	RTH CAROLINA STATE UNIVERSI
Qxbranch	Ranjan Pal	Institute of Technology and Sc	STITUTE OF TECHNOLOGY AND :
Work Fusion	John Elton	Lehigh University	LEHIGH UNIVERSITY
Work Fusion	Upal Basu	Imperial College London	IMPERIAL COLLEGE LONDON

Figure 2: Demonstration of data scraped

4 Automated scoring system construction

With the corresponding education background of employees from different A.I. companies from LinkedIn, we construct an automated rating system to do data analysis. The automated rating system have three parts, the first part is to build a scoring database for as more as possible universities in the world with Python, and the second part is to match the employee database and the scoring database with Excel Fuzzy Lookup and VBA, and the third part is to summarize them at company level.

A. Scoring Database Construction

The criteria of scoring database is that we gather the past 3 years *QS World University Rankings* database(1,000 universities for each year), and take weighted scores of them. For different education level, i.e., bachelor degree, master degree, and doctor degree, we have different quotiety such as:

$$Overall\ Score = Max(a1 * Bachelor\ Score, a2 * Master\ Score, a3 * Doctor\ Score)$$

In the initial version, we set a1 = a2 = 1, a3 = 1.5 for further comparable use.

	Institution Name	Weighted Score
0	MASSACHUSETTS INSTITUTE OF TECHNOLOGY (MIT)	100
1	STANFORD UNIVERSITY	100
2	HARVARD UNIVERSITY	100
3	CALIFORNIA INSTITUTE OF TECHNOLOGY (CALTECH)	100
4	UNIVERSITY OF OXFORD	100
5	UNIVERSITY OF CAMBRIDGE	100
6	ETH ZURICH (SWISS FEDERAL INSTITUTE OF TECHNOL...	97
7	IMPERIAL COLLEGE LONDON	100
8	UNIVERSITY OF CHICAGO	100
9	UCL (UNIVERSITY COLLEGE LONDON)	97
10	NATIONAL UNIVERSITY OF SINGAPORE (NUS)	97

Figure 3: Preview of the Scoring System

B. Automated Matching Model

Since the institution names in the employee database and the names in the scoring system are not strictly the same, which results from uppercase/lowercase mismatch, abbreviation mismatch, etc., we need to find a fuzzy matching method to map the two database. The Fuzzy Lookup is the built-in method of Microsoft Excel, we use Excel VBA for model automation.

Algorithm1: Fuzzy Lookup Automation

Input: Employee, Bachelor, Master, Doctor, Scoresystem(Instituation Name, Score)

```
For i = 2 To 4
    Application.CommandBars("Fuzzy Lookup").Visible = True
    CommandBars.FuzzyLookup(Columns(i), Columns(5))
    For Export New Column:
Application.VLOOKUP(lookup_value, table_array=Scoresystem, column_index=2,
range_lookup=False)
    Next
```

Matching the two database we get each employee scores at bachelor, master, doctor, and overall four level. And we only consider the overall score in the final result comparison.

	Bachelor Institution	Institution Name	Similarity
0	Università di Pavia	UNIVERSITÀ DEGLI STUDI DI PAVIA (UNIPV)	0.900000
1	Beijing Foreign Studies University	BEIJING FOREIGN STUDIES UNIVERSITY	1.000000
2	Politecnico Milano	POLITECNICO DI MILANO	0.950000
3	Queensland University of Technology	QUEENSLAND UNIVERSITY OF TECHNOLOGY (QUT)	0.922222
4	Harvard University	HARVARD UNIVERSITY	1.000000
5	Ecole Polytechnique	ECOLE POLYTECHNIQUE	1.000000
6	University of New South Wales	THE UNIVERSITY OF NEW SOUTH WALES (UNSW)	0.926667
7	Dartmouth College	DARTMOUTH COLLEGE	1.000000
8	Lehigh University	LEHIGH UNIVERSITY	1.000000
9	Vanderbilt University	VANDERBILT UNIVERSITY	1.000000
10	Universite de Montreal	UNIVERSITÉ DE MONTRÉAL	1.000000

Figure 4: Preview of Fuzzy Look Up Result

C. Automated Subtotal Model

Since we consider the employee's performance of a whole company, we consider the employees' scores at a subtotal level for each company, which can also be performed with Excel VBA.

Algorithm 2: Subtotal Automation

```
Columns(Company, Employee, OverallScore).Select
Selection.Subtotal GroupBy:=1, Function:=xlSum, TotalList:=Array(OverallScore)
```

	A	E	H	K	L
1	Company	Bachelor Score	Master Score	Doctor Score	Overall Score
2	Grand Average	83.22	86.56	118.85	89.87
3	Qxbranch Average	84.67	86.27	117.75	89.08
23	Work Fusion Average	80.35	83.86	142.50	85.40
48	Osaro Average	91.21	93.67	110.00	97.00
65	AnotherBrain Average	84.40	89.89	112.50	91.77
77	Deep Instinct Average	78.56	83.50	112.50	82.96
90	Yewno Average	80.86	87.50	125.25	92.35
101	AI Music Average	80.56	82.40	120.50	94.45

Figure 5: Result of Each Company Score

After we get the score table for each company, we can do further one-way and two-way analysis about the employee's major, institution location, etc., in the next part.

5 Pdf resume parsing

We hope to extract the education background of candidate or employee automatically from theirs' resumes in pdf format. In python, we can use PDFMiner to solve this issue. PDFMiner is a tool for extracting information from PDF documents. Unlike other PDF-related tools, it focuses entirely on getting and analyzing text data. PDFMiner allows one to obtain the exact location of text in a page, as well as other information such as fonts or lines. It includes a PDF converter that can transform PDF files into other text formats (such as HTML). It has an extensible PDF parser that can be used for other purposes than text analysis.

Here is the education background part of original resume.

Education

2015 – 2016 **UNIVERSITY OF HONG KONG** **HONG KONG, CHINA**
Master of Science in Information Technology in Education (Specialist Strand: e-Leadership), *Distinction*.
Honorary Career Advisor and Lecturer on VR/AR, Adaptive Learning, Data Analytics, Blockchain, and AI.
Founder of EdTech startup with endorsements from the President of HKU and Dean of Faculty of Education etc.

2009 – 2011 **HARVARD BUSINESS SCHOOL** **BOSTON, MA**
Master in Business Administration. Organizer, HBS Hong Kong Trek. Co-producer, Asian Cultural Show.
Advisor, Harvard Innovation Lab (iLab). Career Advisor, Harvard Business School

2002 – 2006 **UNIVERSITY OF CALIFORNIA, BERKELEY – HAAS SCHOOL OF BUSINESS** **BERKELEY, CA**
Bachelor of Science in Business Administration, *summa cum laude* (cumulative GPA: 3.9, top 3% of class).
Dean's Honor List all semesters. President, California Investment Association (Haas-sponsored investment fund)

Figure 6: Preview of Pdf Resume

Step 1: We build a work folder of all the pdf files that we want to extract. By the pdf_extract.py, we split the text in the pdf into a character format which mainly include four different types of strings

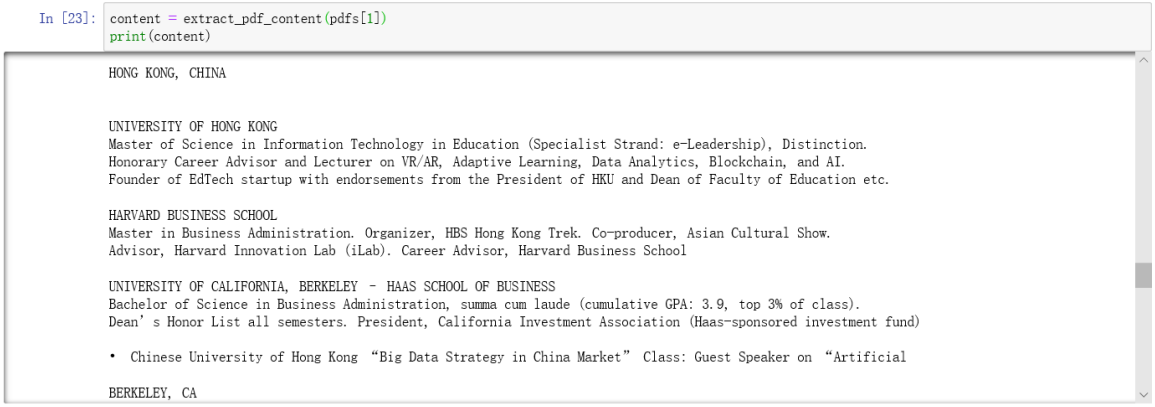
1.English alphabet

2.Space (‘ ’)

3.Line break symbol (‘\n’)

4.Punctuation

Here is an example of our extract output which include all the information of the resume. Although the format is somewhat different from the original resume, the information is not missing.



```
In [23]: content = extract_pdf_content(pdfs[1])
         print(content)

HONG KONG, CHINA


UNIVERSITY OF HONG KONG
Master of Science in Information Technology in Education (Specialist Strand: e-Leadership), Distinction.
Honorary Career Advisor and Lecturer on VR/AR, Adaptive Learning, Data Analytics, Blockchain, and AI.
Founder of EdTech startup with endorsements from the President of HKU and Dean of Faculty of Education etc.

HARVARD BUSINESS SCHOOL
Master in Business Administration. Organizer, HBS Hong Kong Trek. Co-producer, Asian Cultural Show.
Advisor, Harvard Innovation Lab (iLab). Career Advisor, Harvard Business School

UNIVERSITY OF CALIFORNIA, BERKELEY - HAAS SCHOOL OF BUSINESS
Bachelor of Science in Business Administration, summa cum laude (cumulative GPA: 3.9, top 3% of class).
Dean' s Honor List all semesters. President, California Investment Association (Haas-sponsored investment fund)

• Chinese University of Hong Kong “Big Data Strategy in China Market” Class: Guest Speaker on “Artificial

BERKELEY, CA
```

Figure 7: Preview of Parsing Result(1)

Step 2: In order to extract the education background from all the information, we need to find some important symbols. The first category is the name of school, most schools’ name contains the word like ’University’, ‘College’, ‘School’. In most resumes, school name is a separate line so we combine all strings between two line break symbol which contains the word ’University’, ‘College’ or ‘School’ and add it to a list. The second category is the degree, our key words include ‘Phd’, ‘Master’, ‘MBA’, ‘Bachelor’. When we detect these words, we add it to the same list as before. When we get the degree of bachelor, we stop searching because the education background is enough for most companies.

Here is an example of our extract output.

```
In [43]: sentence

Out[43]: ['HARVARD BUSINESS SCHOOL (HBS) ASIA-PACIFIC RESEARCH CENTER  HONG KONG, CHINA ',
          'UNIVERSITY OF HONG KONG ',
          'Master',
          'HARVARD BUSINESS SCHOOL ',
          'Master',
          'Advisor, Harvard Innovation Lab (iLab). Career Advisor, Harvard Business School ',
          'UNIVERSITY OF CALIFORNIA, BERKELEY - HAAS SCHOOL OF BUSINESS ',
          'UNIVERSITY OF CALIFORNIA, BERKELEY - HAAS SCHOOL OF BUSINESS ',
          'Bachelor']
```

Figure 7: Preview of Parsing Result(2)

Step 3: compare all the sentence which contains the word ’University’, ‘College’ or ‘School’ with the name list of university to exclude the extra information to get final result.

Conclusion and further study:

The result above contains all the education background information in the resume, though there are some extra texts. This is a feasible method to extract education

background from resume.
The method has some areas need to be improved.
1.Some people use full name of their schools and others use shorthand, it causes miss match of school.
2.Uppercase and lowercase letters are different in our identification, different people have different habits when writing resumes which cause information missing.
3.The description of the school experience may contain some key words in step2 so it is difficult to decide which is the information we need to reserve.

6 Conclusion and challenges

Through relevant methods and steps, we obtained the data and completed the above analysis and visualization, so that our project can completely capture, identify, integrate the data of personal resume, and complete relevant analysis.
But at the same time, our project also has certain problems and challenges. The first is the anti scraping mechanism, which leads us can not obtain data effectively. Secondly, there are still shortcomings of pdf parsing.

References

[1] Luiza Sayfullina, Eric Malmi, Yiping Liao, Alexander Jung. Domain Adaptation for Resume Classification. Using Convolutional Neural Networks. Part of the Lecture Notes in Computer Science book series (LNCS, volume 10716)