Deep Learning-Based Automatic Entity Extraction and Analysis for Data Science Recruitment: A Postprint

Authors: Wang Dongbo, Haotian Hu, Zhou Xin, Zhu Danhao

Date: 2023-08-27T00:00:00+00:00

Abstract

[Purpose/Significance] Data science, as an emerging interdisciplinary field that integrates numerous domains, is rapidly taking shape. Extracting relevant entity knowledge from data science job postings not only helps understand the development trends of data science from a market perspective, but also contributes to improving the content of data science education. [Method/Process] Based on job postings from major recruitment websites, and combining data acquisition, annotation, and organization methods from information science, we construct a data science recruitment corpus and extract relevant entities from it for analysis and research. [Results/Conclusion] Based on 11,000 annotated job posting corpora collected, we compare the performance of Bi-LSTM-CRF, CRF, and Bi-LSTM models on the task of extracting data science recruitment entities, determine the final automatic extraction model for data science recruitment entities, design an automatic extraction platform for data science recruitment entities, and construct a data science recruitment entity network.

Full Text

Preamble

Volume 62, Issue 13, July 2018
ChinaXiv Partner Journal

Research on Automatic Extraction and Analysis of Data Science Recruitment Entities Based on Deep Learning

Wang Dongbo[1], Hu Haotian[1], Zhou Xin[2], Zhu Danhao[3]
[1]College of Information Science and Technology, Nanjing Agricultural University, Nanjing 210095
[2]School of Information Management, Nanjing University, Nanjing 210093

[3]Department of Computer Science and Technology, Nanjing University, Nanjing 210093

Abstract

[**Purpose/Significance**] Data science is rapidly emerging as a new interdisciplinary field that integrates numerous domains. Extracting entity knowledge from data science recruitment announcements not only helps understand the development trends of data science from a market perspective but also contributes to improving data science teaching content. [**Method/Process**] Based on job postings from major recruitment websites, and combining information science methods for data acquisition, annotation, and organization, this study constructs a data science recruitment corpus and extracts relevant entities for analysis and research. [**Result/Conclusion**] On the basis of 11,000 annotated job announcements, this paper compares the performance of Bi-LSTM-CRF, CRF, and Bi-LSTM models for data science recruitment entity extraction tasks, determines the final automatic extraction model, designs an automatic extraction platform for data science recruitment entities, and constructs a data science recruitment entity network.

Classification Number: G255.1
Keywords: data science, conditional random fields, deep learning, Bi-LSTM-CRF

Data science is rapidly emerging as a new interdisciplinary field that integrates computer science, statistics, applied and computational mathematics, artificial intelligence, systems science, social sciences, psychology, economics, and many other domains. Along this development trajectory, positions such as data scientists, data analysts, data annotators, and data engineers have emerged in large numbers. Acquiring job requirements related to data science and extracting entities such as job titles, required majors, educational requirements, experience requirements, skill levels, and programming languages from unstructured recruitment information through machine learning strategies serves two purposes: first, it helps data science professionals understand market demands for data science talent, enabling them to enhance their capabilities in targeted ways; second, it benefits data science educators in formulating educational systems and talent cultivation objectives.

Current research on English entity extraction has achieved satisfactory results internationally. English entities primarily cover categories such as person names, locations, organizations, time expressions, numbers, and currency, with extraction tasks completed through rule-based, statistical, and machine learning strategies. M.M. Bikel et al. [1] designed methods for extracting person names, locations, and organization names based on Hidden Markov Models, ensuring that entity recognition models built with maximum entropy could select features related to sequences, thereby guaranteeing that both precision and recall outperformed Hidden Markov Models. A.L. Berger et al. [2] proposed a practical

and effective extraction method based on maximum entropy models. J. Lafferty et al. introduced the Conditional Random Fields (CRF) model [3], which combines the advantages of Hidden Markov Models and maximum entropy models, ensuring outstanding performance on entity recognition tasks. M.C. Callum et al. [4] applied the CRF model to automatic entity extraction and verified that its recognition performance was superior to both Hidden Markov Models and maximum entropy models. Because CRF can utilize not only left and right boundary features of entities but also incorporate any features beneficial to entity recognition, the overall performance of the constructed entity recognition model is notably enhanced.

Research on Chinese entity extraction started relatively late, making it less advanced than English entity extraction. The differences in lexical and syntactic structures between Chinese and English sentences make Chinese entity extraction more challenging. Zhang Xiaoheng et al. [5] proposed a method for extracting university names based on manual rules, representing typical rule-based entity recognition research that, while not achieving high overall performance, provided detailed analysis of rule distributions for university names. Y. Zhang et al. [6] developed a system for recognizing named entities and their relationships based on memory-based learning algorithms, which organically utilized boundary features of entities from a statistical perspective and offered methodological reference value. Zheng Fengqiang et al. [7] incorporated sememes as features into the maximum entropy model to improve extraction performance, leveraging not only the maximum entropy model's ability to utilize entity boundary features but also integrating semantic knowledge into model construction, demonstrating innovation from a domain knowledge utilization perspective. Chen Yu et al. [8] attempted to use neural network-based methods to extract entities and their relationships, and although neural networks can fully mine entity features, the recognition performance had significant room for improvement. Shao Fa et al. [9] used disambiguation methods and incorporated HowNet and Bayesian classification to extract entities, thereby addressing polysemy issues. From a methodological perspective, this study transformed the entity recognition task into a classification problem and integrated deep semantic knowledge into the classification model, showing strong innovation. Xu Hua et al. [10] completed entity extraction from medical texts using rule-based methods based on segmented and part-of-speech-tagged medical corpora, achieving high overall performance. While not particularly innovative methodologically, this research demonstrated innovation in domain knowledge mining for medical entities.

Research on entity extraction based on deep learning has emerged in recent years, with representative studies including: Feng Yuntian and Zhang Hongjun et al. [11] extended neural network language models using deep belief networks and proposed a deep architecture for named entity recognition, which provides both macro-level guidance and methodological direction. C. Dong and J. Zhang et al. [12] first applied character-level Bi-LSTM-CRF neural structures to Chinese named entity recognition, achieving good results on the third SIGHAN Bakeoff MSRA dataset, validating the advantages of the Bi-LSTM-CRF combi-

nation and laying a solid foundation for character-based Chinese entity recognition. Zhu Danhao and Yang Lei et al. [13] redefined the input and output for organization name annotation based on RNN methods, proposing a character-level recurrent network annotation model, which was the first application of deep learning to organization entity recognition and holds methodological significance.

Regarding China's specific situation and data science development, domestic researchers have conducted multi-angle explorations: Ye Ying and Ma Feicheng [14] pointed out that data science and information science share the same theoretical logic and technical methods, revealing that data science continues to maintain the basic principles of information science, providing solid theoretical support for data science development from a theoretical perspective. Yang Jing and Wang Xiaoyue et al. [15] analyzed the challenges big data brings to data science analysis tools and introduced emerging big data analysis tools and their development trends, pointing out directions for data science analysis tool development from a big data perspective. Zhou Aoying and Qian Weining et al. [16] discussed the inevitability of data science and engineering as an emerging interdisciplinary field, elaborating on its disciplinary characteristics, knowledge system, and construction 思路, enriching the connotation and extension of data science from an engineering perspective. Chao Lemen and Lu Xiaobin [17] proposed that data science will become the new theoretical foundation for knowledge in the information science field, clarifying the relationship between data science and information science by placing data science within the broader framework of information science. Wang Yuefen and Xie Qingnan et al. [18] conducted a bibliometric analysis of foreign literature on data science using the Web of Science Core Collection database, systematically and comprehensively summarizing domestic and international research on data science from the perspective of bibliometrics, providing first-hand materials for understanding data science development trends and offering references for future research in China.

Building on the above research, this study targeted major domestic recruitment websites, set keywords related to data science, crawled 29,460 job announcements, manually annotated 11,000 data science recruitment announcements, and constructed a Chinese data science recruitment corpus. Based on this corpus, by testing CRF and deep learning models, this study constructed an automatic entity extraction model for data science recruitment announcements, built a corresponding platform, and further analyzed entity distribution patterns using complex networks.

2 Data Science Recruitment Corpus Introduction and Entity Definition

Based on crawling, cleaning, annotating, and organizing data science-related job postings from websites such as Zhaopin and 51job, this study constructed a data science job recruitment corpus through the following process:

(1) Data science job recruitment data was primarily collected from recruitment websites. Using a Python-developed web crawler, 29,460 job announcements were crawled from March 2017 to August 2017.

(2) Based on the crawled data, the job description field was selected to extract 29,460 pieces of recruitment description information. On one hand, through deduplication algorithms, automatic deduplication of the 29,460 announcements was completed, yielding 24,460 deduplicated recruitment announcements. On the other hand, some English recruitment announcements existed in the descriptions. Since this study focuses on Chinese recruitment information, these English announcements were also cleaned, resulting in 23,154 data science recruitment announcements.

(3) Data science recruitment entities in this study refer primarily to entities involved in recruitment details such as job titles, educational requirements, experience requirements, skill levels, and related software. Specific job titles include "software engineer, data analyst, database engineer," etc.; specific educational requirements include "bachelor's degree or above, college degree or above," etc.; specific software includes "MySQL, Python, Java, Spark, SAS, SPSS, R," etc. The concept of data science recruitment entities is defined by adapting the general concept of entities to the specific domain of data science recruitment. Based on the five categories of data science entities identified above, corresponding annotation rules were established, and 55 annotators completed the annotation of entities in 11,000 data science recruitment announcement texts. A sample annotation is shown in Figure 1 [Figure 1: see original paper].

3 Machine Learning Model Introduction

This section provides a brief introduction to the Conditional Random Fields (CRF) model, Long Short-Term Memory (LSTM) model, and LSTM-CRF model.

3.1 CRF Model

Conditional Random Fields are a relatively new model for solving sequence labeling problems. They are undirected graphical models that calculate the joint conditional probability distribution of state labels for an entire observation sequence given a set of observation sequences requiring labeling. The topological structure is shown in Figure 2 [Figure 2: see original paper].

Let $x = \{x_1, x_2, \dots, x_{n-1}, x_n\}$ represent the observed input data sequence, and $y = \{y_1, y_2, \dots, y_{n-1}, y_n\}$ represent the finite set of states, where each state corresponds to a label. Given the input sequence x, the conditional probability of the state sequence y for a linear-chain CRF with parameters $\lambda = \{\lambda_1, \lambda_2, \dots, \lambda_{n-1}, \lambda_n\}$ is:

$$p(y|x,\lambda) = \frac{\exp\left(\sum_{i=1}^{n}\sum_{j}\lambda_j f_j(y_{i-1},y_i,x,i)\right)}{z_x = \sum_y \exp\left(\sum_{i=1}^{n}\sum_{j}\lambda_j f_j(y_{i-1},y_i,x,i)\right)}$$

where z_x is the normalization factor representing the score of all possible state sequences, ensuring that the sum of conditional probabilities for all possible state sequences equals 1. $f_j(y_{i-1}, y_i, x, i)$ is a unified feature function, typically a binary indicator function, and λ_j is the weight of the corresponding feature function obtained after training the model on training data. In building the data science entity recognition model, this model can utilize not only left boundary features of entities but also right boundary features, ensuring that the overall performance of the constructed model is superior to Hidden Markov Models and maximum entropy models.

3.2 LSTM Model

Recurrent Neural Networks (RNN) address the lack of feedback mechanisms in feedforward neural networks when processing continuous sequence inputs by connecting hidden layers. The input set $\{x_0, x_1, \dots, x_t, x_{t+1}, \dots\}$ is treated as an input vector sequence that returns another vector sequence output set $\{y_0, y_1, \dots, y_t, y_{t+1}, \dots\}$. At time t, the RNN hidden layer and output layer are calculated as:

$$h_t = f(Ux_t + Wh_{t-1}) \quad (3)$$

$$y_t = g(Vh_t) \quad (4)$$

In equations (3) and (4), x represents the input layer, h the hidden layer, and y the output layer. U, W, and V are the weights from the input layer to the hidden layer, between consecutive hidden layers, and from the hidden layer to the output layer, respectively. f and g are nonlinear activation functions, typically sigmoid and softmax. Although RNN can theoretically learn long-term dependencies, practical performance is often poor. Long Short-Term Memory (LSTM) networks were proposed to address this issue. LSTM combines a memory cell with gate controllers to control the retention and discarding of historical information. The LSTM memory cell calculations are as follows:

$$i_t = \sigma(W_i h_{t-1} + U_i x_t + b_i) \quad (5)$$

$$f_t = \sigma(W_f h_{t-1} + U_f x_t + b_f) \quad (6)$$

$$o_t = \sigma(W_o h_{t-1} + U_o x_t + b_o) \quad (7)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_c h_{t-1} + U_c x_t + b_c) \quad (8)$$

$$h_t = o_t \odot \tanh(c_t) \quad (9)$$

The activation function σ generally selects the sigmoid function, and $\odot$ represents element-wise multiplication. In equations (5), (6), and (7), i_t, f_t, and o_t represent the input gate, forget gate, and output gate at time t, respectively. In equation (8), c_t represents the memory cell vector at time t. U_i, U_f, U_c, and U_o are the connection weight matrices between the input sequence $\{x_0, x_1, \dots, x_t, x_{t+1}, ...\}$ and each control gate, as well as between the control gates and hidden state h. Since this study conducts character-based data science named entity recognition, during model training, it is necessary to consider not only the connection between the current character and previous context but also incorporate subsequent information for prediction and sequence labeling tasks. Bidirectional LSTM (Bi-LSTM) has two parallel layers in opposite directions that can store information from both directions. Therefore, this study selects Bi-LSTM for entity annotation tasks.

3.3 LSTM-CRF Model

Although LSTM networks can achieve good entity labeling results, when strong dependencies exist between output labels, LSTM model performance is affected. Particularly in actual sequence labeling tasks, the neural network structure's heavy dependence on data means that data volume and quality significantly impact model training effectiveness. To address this issue, this study adopts the LSTM-CRF model. The LSTM-CRF model not only retains LSTM's ability to simultaneously consider contextual information of data science entities but also uses the CRF layer to consider dependencies between independent output labels. Figure 3 [Figure 3: see original paper] shows the LSTM-CRF model structure for entity recognition.

Under the LSTM-CRF model, the output is no longer independent labels but the optimal label sequence. For input $X = \{x_1, x_2, \dots, x_n\}$, we can define A as the state transition matrix and P as the probability matrix output by LSTM. Here, $A_{i,j}$ represents the probability of transitioning from state i to state j in the sequence, and $P_{i,j}$ represents the probability that the i-th character of the input sequence is labeled as the j-th tag. By finding the maximum $s(X, y)$, we can obtain the optimal output label sequence, which is then calculated using dynamic programming to determine the optimal path and perform annotation. The prediction output formula for the label sequence $y = \{y_1, y_2, \dots, y_n\}$ is:

$$s(X, y) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i, y_i} \quad (10)$$

4 Entity Recognition Experiments

4.1 Corpus Preprocessing

Based on the actual length distribution of entities in the manually annotated data science recruitment positions, this study determined to use a 4-tag labeling

set across different models. The labeling set is denoted as R, specifically $R = \{B\text{-}et, I\text{-}et, E\text{-}et, O\}$, where $B\text{-}et$ represents the beginning character of a data science entity, $I\text{-}et$ the middle character, $E\text{-}et$ the ending character, and O characters outside entities. If a data science entity length exceeds 3, $I\text{-}et$ is used for extended characters. This study wrote Python programs to automatically annotate all corpora at the character level based on the " [] " markers for data science entities.

Since GPU is required for deep learning-based entity recognition training, the experimental environment is described as follows: CPU: Intel(R) Core(TM) i5-4590 CPU @ 3.30GHz; Memory: 16GB DDR4; GPU: NVIDIA Quadro K1200; Video Memory: 4GB GDDR5; Operating System: Ubuntu 16.04. The high-performance GPU on the server can support large-scale parallel computing.

4.2 Entity Recognition Evaluation Criteria

This study uses three metrics to evaluate data science entity recognition model performance: precision, recall, and F-measure. The specific calculation formulas are:

$$\text{Precision: } P = \frac{A}{A+B} \times 100\% \quad (11)$$

$$\text{Recall: } R = \frac{A}{A+C} \times 100\% \quad (12)$$

$$\text{F-measure: } F = \frac{2 \times P \times R}{P+R} \times 100\% \quad (13)$$

where A represents the number of correctly recognized data science entities, B the number of incorrectly recognized entities, and C the number of unrecognized entities. It should be noted that accuracy cannot accurately reflect model quality, so this metric is not used.

4.3 Entity Recognition Performance Analysis

This study used CRF, Bi-LSTM, and Bi-LSTM-CRF models for data science entity recognition based on 11,000 manually annotated data science recruitment announcements. Ten-fold cross-validation was used to test model performance, with the 11,000 annotated documents divided into training and test sets at a 9:1 ratio. The test results are shown in Tables 1-3 .

As shown in Table 1 , the character-based data science entity recognition model built with CRF achieved an average F-measure of 85.71%. This F-measure demonstrates that the CRF model can fully utilize left and right boundary character features of data science entities and incorporate these features into model construction. From specific recognition results, professional names, software, and model names were recognized well overall, but entities with ambiguous boundaries were prone to recognition errors or omissions. For example, in the

phrase “对专业数据【分析】及做好竞争对手数据的【采集】、统计、评估与【分析】，并【编制报表】” (for professional data [analysis] and competitor data [collection], statistics, evaluation and [analysis], and [report preparation]), the model accurately segmented “数据” (data) from “分析” (analysis) and “采集” (collection), but failed to recognize “统计” (statistics) and “评估” (evaluation) as entities.

As shown in Table 2 , due to Bi-LSTM's characteristic of having two parallel layers in opposite directions, this feature ensures the precision of data science entity recognition. Compared with the CRF-based model, the data science entity model built with Bi-LSTM improved precision by an average of 1.43%, indicating superior performance. However, when strong dependencies exist between output labels, the LSTM model's performance is affected. In specific examples, for the same phrase mentioned above, Bi-LSTM not only accurately segmented “数据” from “分析” and “采集” but also correctly recognized “统计” and “评估”—entities that CRF and Bi-LSTM models failed to identify.

As shown in Table 3 , the Bi-LSTM-CRF model demonstrates overall good performance for data science entity recognition, with both precision and recall exceeding 90% across all groups. This fully reflects that this combined model not only retains LSTM's ability to consider contextual information simultaneously but also uses the CRF layer to consider dependencies between independent output labels, thereby ensuring both precision and recall for data science recognition models. In specific examples, the Bi-LSTM-CRF model's F-measure ranged from 90.47% to 91.49%, with an average of 91.10%. Overall, it outperformed Bi-LSTM by 3.97% in F-measure, demonstrating that incorporating CRF features effectively improves the entire sequence model's performance. Compared with CRF, Bi-LSTM-CRF's average F-measure was 5.39% higher, fully demonstrating that at the character level, deep learning models can fully leverage end-to-end training and large-scale corpus scenarios. The deep learning model's superiority is also evident in Bi-LSTM's average 1.42% improvement over CRF. In summary, based solely on characters as the fundamental element of Chinese, without any manual feature engineering, the constructed Bi-LSTM-CRF entity recognition model achieves applicable performance levels, providing valuable reference for similar sequence labeling tasks.

4.4 Building an Automatic Extraction Platform for Data Science Recruitment Entities

The data science recruitment entity extraction experiment involves complex steps, such as generating tokens in line format recognizable by Bi-LSTM-CRF and creating corresponding feature templates. After training and testing the corpus, it is also necessary to calculate the three evaluation metrics: precision (P), recall (R), and F-measure. To facilitate experimental operations and help readers understand, this study developed a visual operating system and built an automatic extraction platform for data science recruitment entities by calling the optimal Bi-LSTM-CRF model.

The platform was developed using Python's third-party toolkit PyQt. PyQt, developed by P. Thompson, is a GUI programming solution for Python that successfully integrates Python with the Qt library. PyQt implements a set of Python modules with over 300 classes and nearly 6,000 functions and methods. It is a multi-platform toolkit that can run on all major operating systems, including UNIX, Windows, and Mac. Compared with wxPython and Tkinter, PyQt is more powerful and allows convenient UI design using "Designer" or "QtCreator," simplifying UI layout work.

The platform consists of two main components: (1) data acquisition and cleaning functions, including web crawling and dirty data cleaning; (2) entity extraction and statistical functions, including corpus selection, entity extraction, and frequency statistics.

When using data acquisition and cleaning functions, users first select the desired recruitment announcement publication time range from a dropdown box, with options including: within 24 hours, recent 3 days, recent 1 week, recent 1 month, and all time. After selecting the time range, clicking the "Get Data" button activates the platform's web crawler to grab relevant recruitment announcements from recruitment websites, displaying the crawling progress in a prompt box, as shown in Figure 4 [Figure 4: see original paper]. After announcement crawling is complete, the platform automatically cleans the data and saves all corpora to a specified path.

When using entity extraction and statistical functions, clicking the "Browse" button allows users to select the corpus library (corpus root directory) in the folder browser view, and the system automatically reads all document paths within the corpus library. After clicking the "Extract Entities" button, the platform preprocesses all corpora in the library and generates a text document named "test" in token format per line as required by Bi-LSTM-CRF. It then automatically calls the Windows command prompt (cmd) program to invoke the data science recruitment entity automatic extraction model for entity extraction from the test document, displaying all extracted data science recruitment entities in the "Information Prompt Box," as shown in Figure 5 [Figure 5: see original paper]. Clicking the "Frequency Statistics" button enables the platform to perform frequency statistics on entities extracted by the model and display data science entity frequencies in descending order in the "Information Prompt Box," as shown in Figure 6 [Figure 6: see original paper].

5 Network Analysis of Data Science Recruitment Entities

Based on the Bi-LSTM-CRF extraction model constructed from 11,000 data science recruitment announcements, this study used the automatic extraction platform to extract entities from 12,154 newly crawled data science recruitment announcements. After manual-assisted proofreading, 23,154 data science recruitment entity extractions were completed. Based on entity distribution patterns and entity co-occurrence, this study found that certain connectivity

exists between data science recruitment entities, with a certain scale of entities forming an effective network. Accordingly, this study constructed a data science recruitment entity network.

A primary function of the data science recruitment entity network is to identify main concerns of data science positions—by examining network nodes to discover entities commonly focused on in data science positions, primarily obtained through betweenness centrality in the data science recruitment entity network.

The concept of betweenness centrality was first used to analyze the importance of individuals in social networks, proposed by L.C. Freeman [19] in 1979. He argued that if a node lies between multiple pairs of nodes, its degree might be low. That is, from a degree perspective alone, one might mistakenly believe this node does not occupy a prominent position in the network. However, this low-degree node may play an important role in controlling internal network communications, making it a significant node in the network. Therefore, betweenness centrality can reflect a node's importance in the network, showing the degree of dependence other nodes have on it. For a node i in the network, its betweenness centrality is calculated as:

$$C_B(i) = \sum_{j<k} \frac{g_{jk}(i)}{g_{jk}} \quad (14)$$

where $g_{jk}(i)$ is the number of shortest paths between node pair j and k that pass through node i, and g_{jk} is the total number of shortest paths connecting nodes j and k. Thus, $g_{jk}(i)/g_{jk}$ represents the proportion of shortest paths between nodes j and k that pass through i relative to the total number of shortest paths between j and k.

Betweenness centrality characterizes a node's importance in the network and reflects its ability to control internal network communications. The larger a node's betweenness centrality, the closer its position is to the center of the entire network, meaning the more prominent its position. Because this node is relatively central, large amounts of information in the network will pass through it, making it crucial for controlling internal network communications and particularly important itself. Additionally, betweenness centrality can reflect the centralization degree of the entire network—a key indicator of whether a complex network is mature. If the network's overall betweenness centrality is high, it indicates the network has reached a relatively high level of maturity, presenting a stable and mature state.

Due to the large overall network size, making full display difficult, this study presents two small-scale networks based on 100 documents from the data science corpus. The .net format files were imported into Pajek software to draw data science entity network examples. Figure 7 [Figure 7: see original paper] shows a comprehensive data science entity network composed of educational

requirements, major requirements, experience requirements, and skill requirements. Figure 8 [Figure 8: see original paper] shows a single data science entity network composed only of software entities.

Based on the constructed data science entity network, this study used software entities as analysis samples and selected the top 20 data science software entities ranked by descending betweenness centrality—representing important concerns in data science recruitment software entities. Table 4 presents these top 20 data science software entities sorted by betweenness centrality.

Table 4: Top 20 Data Science Software Entities by Betweenness Centrality

Betweenness Rank	Data Science Software Entity	Betweenness Centrality
1	SQL	0.1429695540
2	Oracle	0.1305702550
3	MySQL	0.1045620190
4	Hadoop	0.0843379040
5	Excel	0.0842922990
6	Python	0.0767000540
7	Linux	0.0643107090
8	Spark	0.0616286510
9	Office	0.0385161950
10	SQL Server	0.0338845110
11	MATLAB	0.0313447170
12	C	0.0275843660
13	R	0.0241636320
14	SAS	0.0240613100
15	SPSS	0.0209335340
16	PPT	0.02000576790
17	Java	0.0193000400
18	Hive	0.0162309970
19	Shell	0.0152836320
20	C++	0.0131819250

As shown in Table 4 , filtering by betweenness centrality values greater than 0.1 reveals that the top three software entities are “SQL,” “Oracle,” and “MySQL.” These three entities are either databases or standard statements for database operations—the foundation and prerequisite for data analysis and mining. Their high ranking fully demonstrates that data storage and retrieval must be completed first for data analysis or mining, with “SQL” ranking first confirming this point. Big data has developed rapidly in recent years, and related technologies are also well-represented in data science recruitment. Among the top 20 software entities, big data-related technologies include “Hadoop” and “Spark,” with “Hadoop” ranking fourth. This data demonstrates the close relationship

between data science and big data and suggests that data science curricula should incorporate more big data technology-related content.

Programming language-related entities in the top 20 mainly include “Java,” “C,” “Python,” “R,” and “MATLAB.” “Python” is a rising star among programming languages in data science recruitment because it is particularly suitable for processing data, especially unstructured data. Therefore, both classroom learning and vocational training should strengthen instruction in this programming language in conjunction with specific data processing tasks. Data science has innumerable connections with statistics, which supports the entire framework and system of data science to a certain extent. The inclusion of “SPSS” and “SAS” among the top 20 entities fully demonstrates this point. Although Office is basic office software, it has unique advantages in data processing and presentation. The inclusion of software entities such as “Excel,” “PPT,” and “Office” strongly proves this, as data science positions involve not only technically demanding roles like model building and algorithm design but also include junior data analysts and data annotators—positions with moderate technical difficulty but high demand. The software tools used in these positions mainly concentrate on commonly used tools like “Excel” and “PPT.”

Due to space limitations, this paper only presents the distribution of the top 20 software entities in the data science entity network and analyzes typical software entities in conjunction with corresponding recruitment requirements.

References

[1] Bikel DM, Schwartz R, Weischedel RM. An algorithm that learns what’s in a name[J]. Machine learning, 1999, 34(1/3): 211-231.

[2] Berger AL, Pietra VJD, Pietra SAD. A maximum entropy approach to natural language processing[J]. Computational linguistics, 1996, 22(1): 39-71.

[3] Lafferty J, McCallum A, Pereira F. Conditional random fields: probabilistic models for segmenting and labeling sequence data[C]//Proceedings of the eighteenth international conference on machine learning. San Francisco: Morgan Kaufmann, 2001: 282-289.

[4] McCallum A, Li W. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons[C]//Proceedings of the seventh conference on natural language learning at HLT-NAACL. Association for Computational Linguistics, 2003: 188-191.

[5] Zhang Xiaoheng, Wang Lingling. Recognition and analysis of Chinese organization names[J]. Journal of Chinese Information Processing, 1997, 11(4): 21-32.

[6] Zhang Y, Zhou JF. A trainable method for extracting Chinese entity names and their relations[C]//The Workshop on Chinese Language Processing: Held in

Conjunction with the Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2000: 66-72.

[7] Zheng Fengqiang, Lin Lei, Liu Bingquan. Research on the application of HowNet in named entity recognition[J]. Journal of Chinese Information Processing, 2008, 22(5): 97-101.

[8] Chen Yu, Zheng Dequan, Zhao Tiejun. Chinese named entity relation extraction based on Deep Belief Nets[J]. Journal of Software, 2012, 23(10): 2572-2585.

[9] Shao Fa, Huang Yinge, Zhou Lanjiang, et al. Chinese entity relation extraction based on entity disambiguation[J]. Journal of Shandong University (Engineering Science), 2014, 44(6): 32-37.

[10] Xu Hua, Liu Maofu, Jiang Li, et al. Entity extraction of disease bacteria based on language rules[J]. Journal of Wuhan University (Natural Science Edition), 2015, 61(2): 51-55.

[11] Feng Yuntian, Zhang Hongjun, Hao Wenning, et al. Named entity recognition based on deep belief networks[J]. Computer Science, 2016, 43(4): 224-230.

[12] Dong C, Zhang J, Zong C, et al. Character-based LSTM-CRF with radical-level features for Chinese named entity recognition[C]//International conference on computer processing of oriental languages. New York City: Springer International Publishing, 2016: 239-250.

[13] Zhu Danhao, Yang Lei, Wang Dongbo. Research on Chinese organization name recognition based on deep learning: A character-level recurrent neural network method[J]. New Technology of Library and Information Service, 2016, 32(12): 36-43.

[14] Ye Ying, Ma Feicheng. The rise of data science and its association with information science[J]. Journal of Intelligence, 2015(6): 575-580.

[15] Yang Jing, Wang Xiaoyue, Bai Rujiang, et al. Current status and development trends of data science analysis tools under the big data background[J]. Information Studies: Theory & Application, 2015, 38(3): 134-138.

[16] Zhou Aoying, Qian Weining, Wang Changbo. Data science and engineering: An emerging interdisciplinary field in the big data era[J]. Big Data Research, 2015, 1(2): 90-99.

[17] Chao Lemen, Lu Xiaobin. Data science and its impact on information science[J]. Journal of Intelligence, 2017, 36(8): 761-771.

[18] Wang Yuefen, Xie Qingnan, Song Xiaokang. Review and prospect of foreign data science research[J]. Library and Information Service, 2016, 60(14): 5-14.

[19] Freeman LC. Centrality in social networks conceptual clarification[J]. Social networks, 1979, 1(3), 215-239.

Author Contributions

Zhou Xin: Performance evaluation and testing;
Wang Dongbo: Proposed the paper framework, designed algorithms, and wrote the paper;
Zhu Danhao: Model training;
Hu Haotian: Data annotation, model parameter adjustment, and paper writing.

Research of Automatic Extraction of Entities of Data Science Recruitment and Analysis Based on Deep Learning

Wang Dongbo[1], Hu Haotian[1], Zhou Xin[2], Zhu Danhao[3]
[1]College of Information Science and Technology, Nanjing Agricultural University, Nanjing 210095
[2]Department of Information Management, Nanjing University, Nanjing 210093
[3]Department of Computer Science and Technology, Nanjing University, Nanjing 210093

Abstract: [Purpose/significance] Data science is emerging as a new interdisciplinary field which combines many fields. Extracting the corresponding entities knowledge from the announcement information of data science recruitment can not only help understand the development dynamics of data science from a market perspective, but also help improve the content of data science teaching. [Method/process] Based on the job recruitment announcements from major recruitment websites, combined with information science data acquisition, annotation and organization methods, a data science recruitment corpus was constructed and the corresponding entities were extracted for analysis and research. [Result/conclusion] On the basis of 11,000 annotated recruitment announcements, based on Bi-LSTM-CRF, CRF and Bi-LSTM models, this paper compares the extraction performance of data science recruitment entities, determines the final automatic extraction model of data science recruitment entities, designs the data science recruitment entity automatic extraction platform, and builds a data science recruitment entity network.

Keywords: data science, conditional random field, deep learning, Bi-LSTM-CRF

Note: Figure translations are in progress. See original paper for figures.