Research on the Design and Implementation of a Knowledge Graph Construction Tool

XueSong He and Yibo Liu(✉)

College of Information and Communication, National University of Defense Technology, Wuhan 430019, China
liuyibo@nudt.edu.cn

Abstract. Since its development, knowledge graph has been widely used in the field of data processing. In this paper, by designing and implementing knowledge graph construction and visualization tools, we aim to solve the problems that the current knowledge graph construction process in general fields is not standardized and the construction cost is high, we adopt the Django framework of python to develop web pages, the secondary graph database stores data, and Echarts visually displays knowledge graph, thus realizing a knowledge graph construction tool with the functions of resource introduction, text annotation, extraction and recognition, and graph overview, and propose a solution for knowledge graph construction with convenient operation and comprehensive functions, so as to show knowledge intuitively in a visual way. In this paper, by comparing with different types of knowledge graph platforms in China, such as Huawei Cloud, KGCloud, and Taoist Knowledge graph, we adopt the Django framework of python to develop web pages, the secondary graph database stores data, and Echarts visually displays knowledge graph, thus realizing a knowledge graph construction tool with the functions of resource introduction, text annotation, extraction and recognition, and graph overview, and supporting manual annotation of triples. The introduction of various types of resource files, entity identification and relationship extraction models has solved the various needs of users in the process of building knowledge graphs, contributed to the popularization of knowledge graphs, and enabled users to process data through knowledge graph technology more conveniently, thus improving the efficiency of data mining.

Keywords: Knowledge graph · Relationship extraction · Named entity recognition · Model import · Tool platform

1 Introduction

With the rapid development of science, the social production structure has undergone major changes, and people have ushered in the era of information revolution. With the rapid development of mobile computing, cloud computing, big data and other technologies, the massive data in political, economic, military, entertainment, medical and other fields have been growing rapidly in many aspects, such as social networks, e-commerce, Internet of Things, etc. In order to make efficient use of these data and fully tap a large

P. Qi and Z. Chen (Eds.): ICBDIE 2023, AISR 178, pp. 605–615, 2024.
https://doi.org/10.2991/978-94-6463-238-5_82

amount of knowledge hidden in the data, a lot of research on knowledge graph has been carried out at home and abroad in order to present the complicated massive data in the form of knowledge according to the needs of users, which symbolizes that the World Wide Web is moving from the "Web 2.0" to the "Web 3.0" [1].

Knowledge graph has strong semantic processing ability and open interconnection ability, and has obvious advantages of man-machine mutual assistance, which makes it possible to realize the knowledge interconnection of "Web3.0". It is a knowledge interconnection method that not only meets the cognitive needs of users, but also conforms to the development and changes of network information resources [2]. Its essence is a huge semantic network, which describes the entities and relationships in the real world through nodes and edges on the network respectively. The concept entities and attributes are effectively displayed by their link relationships, which makes the stored knowledge easier to be recognized and interpreted by computers and reduces the burden of data extraction and calculation in traditional knowledge interaction between man and machine [3]. On the other hand, unstructured and semi-structured data are constructed into knowledge graphs through information extraction, knowledge fusion, knowledge updating and other technologies, so that massive data can be integrated into knowledge that human beings can understand, manage and apply in a structured form, and the expression form conforms to the integrity and relevance of human cognition.

2 Compare with Some Mainstream Knowledge Graph Software

At present, the domestic knowledge graph construction tools are still in the development stage. For the construction of knowledge graph, users can choose the corresponding knowledge graph construction tools according to their own needs, but no mainstream construction tools have been widely used in the field [4]. According to different types, this paper will compare and study three different types of knowledge graph construction tools, namely Huawei Cloud knowledge graph platform developed by Huawei Company, KGCloud, a basic knowledge graph construction tool for network open source, and philosophical knowledge graph software. Among them, Huawei Cloud, KGCloud and the knowledge graph construction tool realized in this paper run on Web pages, and the philosophical knowledge graph runs on software clients.

2.1 Same Software Introduction

Huawei Cloud Knowledge graph is a paid knowledge graph construction tool developed and operated by Huawei. Users need to pay for the platform according to the specifications of the graph and the time of purchase. Its functions include importing data source, importing atlas ontology, information extraction, knowledge graph, knowledge fusion, atlas quality inspection and other modules. Users can import data source files into Huawei Cloud according to their own needs, and integrate data according to information extraction models provided by the platform. At the same time, they can upload local knowledge extraction and entity recognition models to the cloud library, and use ontology models to construct knowledge graphs.

KGCloud knowledge graph is a basic knowledge graph construction tool with open source network, simple interface, simple function and open source code, which is convenient for individuals to learn knowledge graph. The platform includes the functions of importing resources, creating new entities/concepts and relationships/attributes, searching, labeling, atlas overview and knowledge fusion. It is very friendly to individual users. Users can use all the functions of the platform for free, import resources, create new entities and relationships, observe the knowledge graph intuitively, and fuse with other graphs.

The philosophical knowledge graph is an earlier and more comprehensive knowledge graph construction software in China. Its main users are enterprises, and individual users can provide personal information application experience. Its main functions are relatively perfect, including knowledge modeling, knowledge extraction, knowledge fusion, knowledge storage, graph application and other ecology. The functions of data analysis, intelligent search, intelligent recommendation and decision support are realized. Users can directly apply the observed data after completing the construction of knowledge graph.

2.2 Software Comparison

The above briefly introduces three different types of knowledge graph construction tools. By comparing the three tools, we can find that. Huawei cloud knowledge graph construction platform needs to upload user data to Huawei cloud for storage, which is not conducive to users' integration and construction of data with high privacy, and the import process is complicated, which requires a certain operating basis, and the long-term use cost is high, which is not suitable for individual users. The interface of KGCloud knowledge graph platform is simple, but the account security is low and the function is not perfect. Users can only use the simple entity identification and knowledge extraction model provided by the platform, and do not support personal import. The application threshold of philosophical knowledge graph is high, and it is difficult for individual users to apply, and its main function is oriented to enterprises and not suitable for individual users.

Combined with the above comparison, the knowledge graph construction tool studied in this paper aims to propose a website platform with simple operation, perfect functions, giving full play to users' initiative, meeting different needs of users, facilitating users to learn knowledge graphs, and at the same time facing some basic applications and providing basic support for enterprise users.

3 Total Design

3.1 Functional Overview

In order to construct and display the knowledge graph more conveniently, quickly and efficiently, and realize the integration and mining of data by users, this paper compares and studies the existing tools, and puts forward the design and implementation scheme

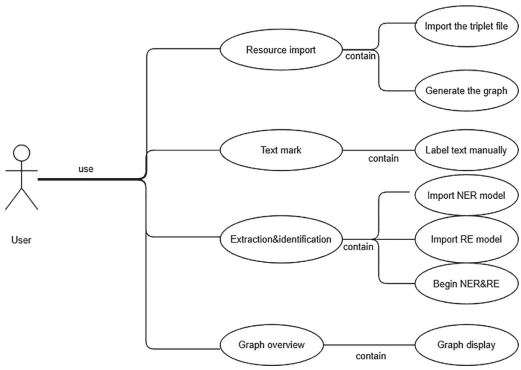

Fig. 1. General use case diagram

of the knowledge graph construction and visualization tool. The tool has four modules: resource import, text annotation, extraction and recognition, and graph overview, which can display data intuitively in graphical form.

The resource import module includes the functions of importing triple files and generating graphs; The text labeling module includes the function of manually labeling text; The extraction identification module includes an import entity model, an import relationship extraction model and an entity and relationship extraction module; The atlas overview module includes atlas display function (Fig. 1).

3.2 Overall Architecture

The knowledge graph construction and visualization tools realized in this paper use python's Django framework to develop the platform, the secondary graph database to process data, and Echarts to visualize the knowledge graph.

The overall framework of the knowledge graph construction and visualization tool realized in this paper consists of three layers, namely, data layer, data processing layer and knowledge graph construction layer. The data layer mainly obtains structured and unstructured data imported by users; The data processing layer mainly carries out knowledge acquisition, knowledge representation and storage of data; The knowledge graph construction layer carries out entity identification on the data, constructs a knowledge graph and displays it visually (Fig. 2).

3.3 Module Design and Implementation

According to the user's functional requirements for the knowledge graph construction tool and the previous research and design, the function of the tool is divided into four modules: resource import, text labeling, extraction and identification, and graph overview (Fig. 3).

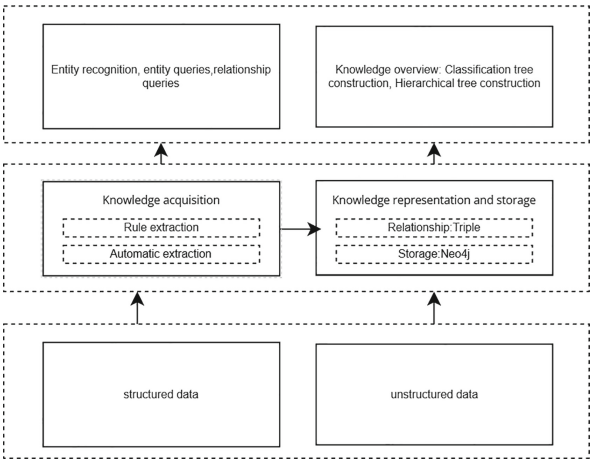

Fig. 2. The overall architecture design of the knowledge graph platform

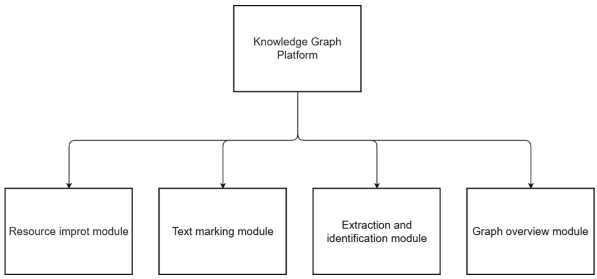

Fig. 3. Functional block diagram of knowledge graph application system

4 Arithmetic

If the user does not import the entity recognition and relationship extraction model, if you need to build a knowledge graph, you can use the default entity recognition and relationship extraction joint model. The following is introduced through three aspects: data labeling strategy, data preprocessing, and model structure.

4.1 Data Labeling Policies

Using the BIO tagging strategy, words in a sentence should be subscripted (token_id), words (tokens), markup labels (BIO), entity relationships (N if no relationship is present) (relationships), and corresponding relationships subscripted.

4.2 Data Preprocessing

i. Read all the data to get the complete set of radicals of words, the chars_set complete set of bios_set of entity labels, the complete set of relationships,relations_set.

ii. Traverse the training data and encapsulate the token_id, token, bio, relations, heads in each sentence as a list into that sentence. Then iterate the current sentence to idize the sample data, embedding_ids the list of words in the sentence, the list of radical id char_ids s, the list of bio_ids, and the list of relationships into the sentence.
iii. Process sentence-idized data so that the dimensions of each sentence in a batch of data are equal, and the dimension of the longest sentence is used as the largest dimension, and the insufficient is filled with 0. Among them, the processing of the scoringMatrix Heads relationship needs to be specially explained, first initialize a [sentence length, sentence length *len(relations_set)] 0 matrix scoringMatrix, traverse scoringMatrixHeads, and fill 1 with the id calculated by step 2 of each word as the column vector of the scoringMatrix matrix, and use 1 to represent the relationship between words.

4.3 Model Structure

i. Word Embedding layer: first initialize the weight parameters of the radical char_ids, embedding the word, and extract the features through bidirectional LSTM to obtain char_logitics. Load the word vector pre-trained by the skip-gram model to obtain word embedding, and use word embedding and char_logitics stitching as input inputs to the model.
ii. Bidirectional LSTM layer: The feature extraction of input inputs by bidirectional LSTM of three hidden layers is lstm_out.
iii. Make a full connection of the lstm_out activation function to relu, classify the entity, and obtain nerScores.
iv. Through the BIO tagging strategy, CRF is used to introduce dependencies between labels. First, the score of each word to get different labels is calculated, then the probability of the label sequence of the sentence is calculated, the ner_loss is obtained by minimizing the cross-entropy loss function, and finally the label with the highest score is obtained by using the viterbi algorithm preNers.
v. The label embedding obtained in step 4 (the real label is used for training, and the predicted label preNers is used for testing) is embedding, and the lstm_out output in step 2 and the label Embedding are spliced to obtain rel_inputs as input for entity relationship prediction.
vi. The following formula is used to calculate the relations and head vectors (i.e., relations and heads in the sample) that each word is most likely to correspond to, and the rel_scores.

The purpose is to identify the most likely head vector $w_i, i \in \{0, ..., n\}$ and the corresponding most likely relationship label $\hat{y} \subseteq$ w for each word in the input word sequence $\hat{r} \subseteq R$, and calculate the scores of r_k, the relationship between token w_i and w_j using the following formula:

$$s^{(r)}(z_j, z_j, r_k) = V^{(r)} f(U^{(r)} z_j + W^{(r)} z_i + b^{(r)}) \tag{1}$$

(r): Relationship extraction
$f(\cdot)$: Activation function, such as relu and tanh;

$$V^{(r)} \in R^l, U^{(r)} \in R^{l \times (2d+b)}, W^{(r)} \in R^{l \times (2d+b)}, b^{(r)} \in R^l$$

d: hidden size of the LSTM
b: size of the label embeddings
l: layer width

The score obtained above is processed by the sigmoid function to obtain the probability that the token w_j is selected as the head having relationship r_k with the token w_i.

vii. Sigmod cross-entropy is performed on the obtained rel_scores and the scoring-Matrix obtained in data preprocessing, and the loss rel_loss is obtained. Sigmod prediction of entity relationships for rel_scores gets pre_Rel.
viii. Adversarial training layer: Adversarial samples are obtained by adding the worst perturbation to the original embedding to maximize the loss function.

Noise data: Loss is derived from the word vector, L2 regularization, and multiplied by a coefficient.

The final loss is obtained using the following formula:

$$\eta_{adv} = \arg\max_{||\eta||} \leq \epsilon^{l_{JOINT(\omega+\eta;\hat{\theta})}} \tag{2}$$

$\hat{\theta}$: $\hat{\theta}$ is copied from the parameters of the current model.

The above formula is difficult to deal with in neural networks, so the approximate method proposed by Goodfellow et al. is adopted:

$$\eta_{adv} = \epsilon\, g/\|g\| \tag{3}$$

$$g = \nabla_{\omega}{}^{l}{}_{JOINT(\omega;\hat{\theta})} \tag{4}$$

ϵ: The small bounds specification, seen as a hyperparameter, is the same as Yasunaga et al., set to $a\sqrt{D}$, where D is the dimension of embedding.

Then the original sample and the adversarial sample are mixed for training, and the final loss function is:

$$l_{JOINT(\omega;\hat{\theta})} + l_{JOINT(\omega+\eta_{adv};\hat{\theta})} \tag{5}$$

5 Experiment and Test

5.1 Login Interface

The login interface consists of a navigation bar and a login window. The login window consists of two buttons: login and registration. Users need to register their accounts for the first time, and then log in through the account password. If login fails, they can retrieve and modify the password through the function of forgetting the password. The navigation bar can search pages and functions and view login information (Fig. 4).

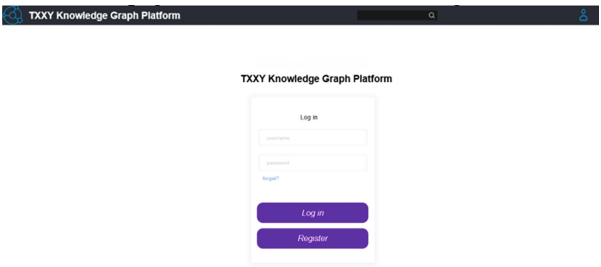

Fig. 4. Login interface

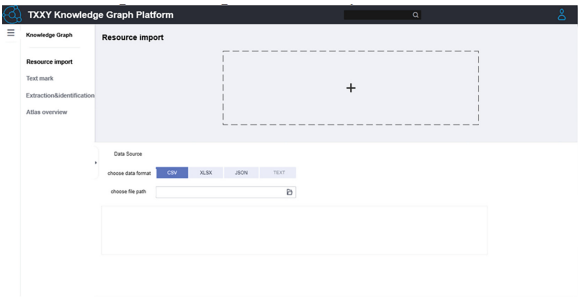

Fig. 5. The interface of resource import module

5.2 Resource Import Module

It mainly has the function of graph construction, and users can import triple files and other resources and generate knowledge graphs. After logging in, users enter the resource import interface by default, which consists of three parts: the menu bar on the left, the resource import bar and the data source bar. Users can drag files into the resource import bar or select data formats such as CSV and JSON in the data source bar and select the data source path to import manually (Fig. 5).

5.3 Text Marking Module

It mainly has the function of text annotation, users can import text and display it in the center of the interface, and users can manually label the text in triples. After importing resources, users click on the text annotation module in the left menu bar to enter the text annotation interface, which consists of two parts: the left menu bar and the text annotation column. It supports manual annotation of triples and displays entities and relationships at the bottom (Fig. 6).

5.4 Extraction and Identification Module

It mainly has the functions of importing entity identification model and relationship extraction model, and users can import entity identification model and relationship

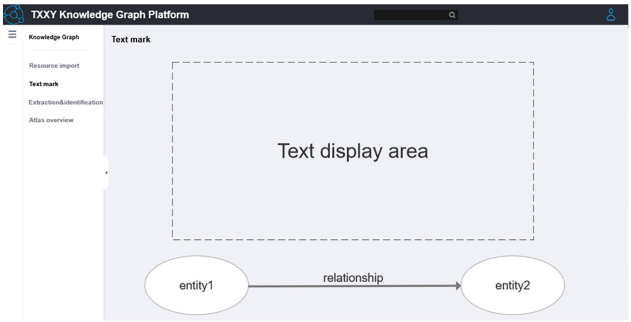

Fig. 6. The interface of text marking module

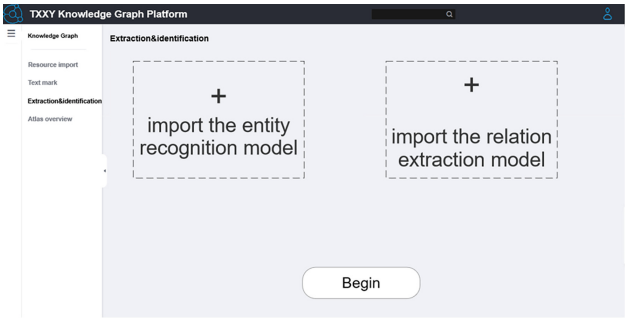

Fig. 7. The interface of extraction and identification module

extraction model to extract entities and relationships. The user enters the extraction identification interface by clicking the extraction identification module in the left menu bar, which consists of two parts: the left menu bar and the extraction identification column. The user can drag the files containing the entity identification model and the relationship extraction model into the corresponding function boxes respectively, and click the Start button to complete the entity identification and relationship extraction (Fig. 7).

5.5 Graph Overview Module

It mainly has the function of graph display, which realizes the visualization of knowledge graph, and users can visually observe knowledge graph in a graphical way. Users click on the graph overview module in the left menu bar to enter the graph overview interface, which consists of two parts: the left menu bar and the graph overview column. The graph overview column shows the knowledge graph composed of concepts, attributes, entities and relationships, and users can visually observe the graphical knowledge graph and input content to search for triples that meet specific conditions (Fig. 8).

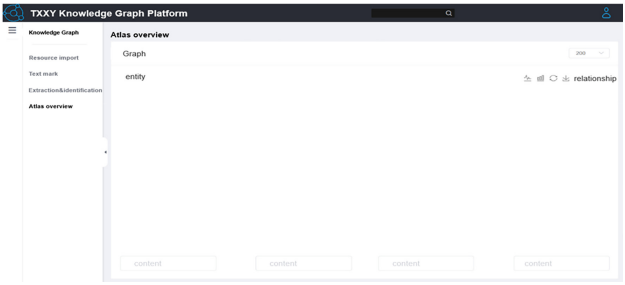

Fig. 8. The interface of Graph overview module

6 Summary

From the application point of view, this paper studies how to construct knowledge graph conveniently and simply. After investigation, it compares with the knowledge graph construction tools of various domestic platforms, and develops the knowledge graph construction tools on this basis, which has a certain promotion effect on the popularization and application of knowledge graph.

From the user's perspective, the knowledge graph construction tool developed in this paper has lowered the user's use threshold, and users can use all the functions in the tool for free. The interface function operation is very simple, and users can get started quickly, which is convenient for learning related concepts of knowledge graph. At the same time, the tool has comprehensive functions and can provide basic service support for enterprise users [5].

From the data level, users can import local data into the platform, without uploading data to the cloud, and rely on tools to complete the knowledge graph construction at the web page. The construction results can be saved locally, which can better protect users' data privacy and prevent users' data from leaking.

Considering from the functional level, the tool contains the main functions in the process of knowledge graph construction [6]. Users can import data source files with different formats, manually label the text with triples, import knowledge extraction models and entity recognition models that meet users' needs, display the knowledge graph graphically, and observe specific content through search and display it intuitively [7].

Although the knowledge graph construction tool developed in this paper has achieved the main functions in the process of knowledge graph construction, it is still not perfect in terms of user's self-selection and the details of the use of some functions [8]. It is necessary to enrich the page design, improve the user-friendliness of the functions, provide the corresponding basic construction model, and learn from the tools such as philosophical knowledge graph to gradually develop application services such as question answering system and intelligent search.

Acknowledgments. This work was guided by Yibo Liu.

References

1. Qi Guilin, Gao Huan, Wu Tianxing. Research progress of knowledge graph [J]. Information Engineering, 2017,3(1):4–25.
2. Huang Yongxiang. Django Web application development [M]. Beijing: Tsinghua University Publishing House, 2019.
3. Cui Peng. Application of ECharts in data visualization [J]. Software Engineering, 2019,22(6):42-46.
4. Buscaldi D, Zargayouna H. Yasemir: Yet another semantic information retrieval system[C]. Proceedings of the sixth international workshop on Exploiting semantic annotations in information retrieval. California: ACM,2013:13–16.
5. Lample G, Ballesteros M, Subramanian S, et al. Neural architectures for named entity recognition[J]. arXiv preprint arXiv:1603.01360, 2016.
6. LI Y, HAO X, WANG Y. N-ary Chinese Open Entity-relation Extraction[J]. Compute rScience, 2017: S1.
7. Wangxin, Zou Lei, Wang Chaokun, Peng Peng, Feng Zhiyong. Summary of knowledge graph data management research. Journal of Software, 2019,30(7):21392174. http://www.jos.org.cn/1000-9825/5841.htm
8. Li Mingxin, Wang Song. Research context and theme analysis of domestic knowledge graph in recent ten years [A]. School of Computer Science and Information Technology, Northeast Normal University, Changchun, 2016.