Data visualization in Network Economics

Zhuobiao Qiao
School of Electronic Information
and Electrical Engineering
Shanghai Jiao Tong University
Shanghai
Email: jdqiaozb@sjtu.edu.cn

Abstract

Today, the crimes of the sale of counterfeit cigarettes have shifted from offline to online development, which brings huge profits for dealers, but also brings huge economic losses and safety hazards for human and countries. Our project aims for finding the hidden relationship of known criminals. Our assignment is crawling data from HTML and XML file, create a web framework for collecting, then drawing a network diagram using the data after desensitization and show the related relationship between the characters. Furthermore, we can find out the social characteristics of gang members so as to realize the identification and prevention of similar crimes. We use sigmajs, a JavaScript library dedicated to graph drawing. It makes easy to public networks on Web pages. For analyzing, we use Gephi to show out the relationship with point and edge.

1. Introduction

China is a big country of tobacco. At present, China's tobacco production tobacco consumption and smoking are the world's first tobacco production of 33 million cases (50 thousand per case), is 4 times the United States second tobacco producing countries, the global tobacco market is 31%; China's smoking number 320 million, the total number of smokers in the world (1 billion 100 million) more than 1/4. Since smoking is a bad behavior which is harmful for human, the government control the sale strictly. At present, the acts to sell cigarettes and cigars to consumers using the Internet are unauthorized violations. According to our survey, the current cigarette sales through the Internet are all counterfeit. The poor quality of cigarettes seriously endangering the health of consumers. So our goals are divided into several part:

1) Extract data from HTML files and XML file.

2) Create a web framework for collecting data.

3) Set up a set of complete, format-unified, actual and diverse database, provide consistent, verifiable, comparable basic data for a variety of judgment platform or analysis tools.

4) The current goal is to generate a million class database that contains more than 100 verifiable relationships and implements associations between different types of data, as well as providing a learning sample for a simple machine learning model.

5) The further goal is to generate a ten million class database that contains more than 1000 verifiable relationships and implements associations between different types of data, as well as providing a learning sample for a more complex machine learning model.

6) The final goal is to generate a hundred million class database that contains more than 10000 verifiable relationships and implements associations between different types of data, as well as providing a learning sample for a more complex machine learning model.

Knowledge graphs model information in the form of entities and relationships between them. This kind of relational knowledge representation has a long history in logic and artificial intelligence [1], for example, in semantic networks [2] and frames [3]. More recently, it has been used in the Semantic Web community with the purpose of creating a web of data that is readable by machines [4]. While this vision of the Semantic Web remains to be fully realized, parts of it have been achieved. In particular, the concept of linked data [5, 6] has gained traction, as it facilitates publishing and interlinking data on the Web in relational form using the W3C Resource Description Framework (RDF) [7, 8]. (For an introduction to knowledge representation, see e.g. [1, 9, 10]).

For the collecting data, it contains two part:

i. HTML file, It has a boot HTML, which embodies the

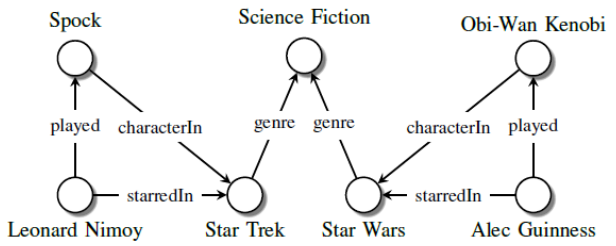


Figure 1: Sample knowledge graph. Nodes represent entities, edge labels represent types of relations, edges represent existing relationships.

whole information a mobile phone have stored. They're also contained in other HTML files, such as address book, telephone message, short message, etc. About other information, it's said that police have achieved them, so we only consider the information about the three things. We should design algorithms to crawl data from HTML file. For XML file, there should be another different algorithm used for extracting data. ii. For the graph, the sample data is twenty-four templates, each template includes ten complete entries (the actual data may be missing some important data), the main data include, but are not limited to:

i. Case data: the focus is case number.

ii. Personnel data: contains basic information about personnel, the key information is ID information and the case number.

iii. Capital data: including banks, Alipay, WeChat, caifutong payment. Note that the bank data mainly includes the balance type and amount of payments, is composed of three kinds of carrying water bank, the Agricultural Bank is the distinction between positive and negative, such as 100.00, -300.00; the other two banks only through the balance type to determine income or expenditure, there is no positive and negative points.

iv. Communication data: including address book (seemingly not important), call records, short messages. Call records including the machine and call time on the end of the two number, call type (Master/called) may produce, two call records identical (such as two times, but not calling) cannot be counted as a data. All need to be reserved when desensitization.

v. Logistics data: it's some tracking numbers and so on.

vi. Verifiable list of relationships: not yet defined, and will be discussed in future.

2. Knowledge Graph

2.1. Knowledge base construction

Completeness, accuracy, and data quality are important parameters that determine the usefulness of knowledge bases and are influenced by the way knowledge bases are constructed. We can classify KB construction methods into four main groups:

i) In *curated* approaches, triples are created manually by a closed group of experts.

ii) In *collaborative* approaches, triples are created manually by an open group of volunteers.

iii) In *automated semi-structured* approaches, triples are extracted automatically from semi-structured text (e.g., infoboxes in Wikipedia) via hand-crafted rules, learned rules, or regular expressions.

iv) In *automated unstructured* approaches, triples are extracted automatically from unstructured text via machine learning and natural language processing techniques (see, e.g., [23] for a review). Construction of curated knowledge bases typically leads to highly accurate results, but this technique does not scale well due to its dependence on human experts. Collaborative knowledge base construction, which was used to build Wikipedia and Freebase, scales better but still has some limitations. For instance, as mentioned previously, the place of birth attribute is missing for 71% of all people included in Freebase, even though this is a mandatory property of the schema [22]. Also, a recent study [24] found that the growth of Wikipedia has been slowing down. Consequently, automatic knowledge base construction methods have been gaining more attention.

2.2. Knowledge Task

The main tasks for final exhibition are including as follows:

i. **Link prediction** is concerned with predicting the existence (or probability of correctness) of (typed) edges in the graph (i.e., triples). This is important since existing knowledge graphs are often missing many facts, and some of the edges they contain are incorrect [11]. In the context of knowledge graphs, link prediction is also referred to as knowledge graph completion. For example, in Figure 1, suppose the characterIn edge from Obi-Wan Kenobi to Star Wars were missing; we might be able to predict this missing edge, based on the structural similarity between this part of the graph and the part involving Spock and Star Trek. It has been shown that relational models that take the relationships of entities into account can significantly outperform non-relational machine learning methods for

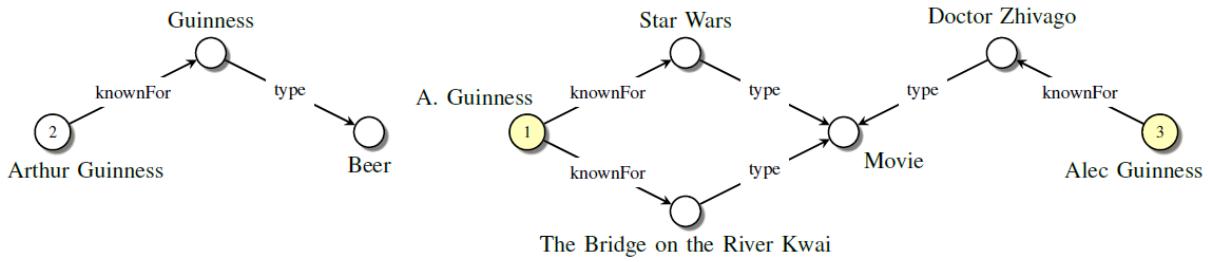


Figure 2: Example of entity resolution in a toy knowledge graph. In this example, nodes 1 and 3 refer to the identical entity, the actor Alec Guinness. Node 2 on the other hand refers to Arthur Guinness, the founder of the Guinness brewery. The surface name of node 2 (A. Guinness) alone would not be sufficient to perform a correct matching as it could refer to both Alec Guinness and Arthur Guinness. However, since links in the graph reveal the occupations of the persons, a relational approach can perform the correct matching.

this task (e.g., see [12, 13]).

ii. **Entity resolution** (also known as record linkage [14], object identification [15], instance matching [16], and deduplication [17]) is the problem of identifying which objects in relational data refer to the same underlying entities. See Figure 2 for a small example. In a relational setting, the decisions about which objects are assumed to be identical can propagate through the graph, so that matching decisions are made collectively for all objects in a domain rather than independently for each object pair (see, for example, [18, 19, 20]). In schema-based automated knowledge base construction, entity resolution can be used to match the extracted surface names to entities stored in the knowledgegraph.

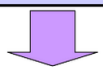
iii. **Link-based clustering** extends feature-based clustering to a relational learning setting and groups entities in relational data based on their similarity. However, in link-based clustering, entities are not only grouped by the similarity of their features but also by the similarity of their links. As in entity resolution, the similarity of entities can propagate through the knowledge graph, such that relational modeling can add important information for this task. In social network analysis, link-based clustering is also known as community detection [21].

3. Related Work

Before starting work, the data from police should have a data masking for non-sensitive. There have been some famous company doing it.

Oracle: In the testing process, the user often need to simulate a certain amount of data, in order to simulate the current production environment more accurately, make the result of testing closer to the actual environment, production data usually needs to be cloned into the test environment. However, for the sake of production safety and compliance, sensitive information such as customer name and certificate number is prohibited from flowing out of production environment. In order to solve the above problem, which make test environment not only close to the production environment, but also it can avoid the leakage of sensitive information, *EM data masking package* provides the function to the user that shielding off one of the sensitive data while retaining production data. For example, change the ID number into a group of unordered numbers, shuffle all the user names in the table, and so on. as shown in fig.3:

LAST_NAME	SSN	SALARY
AGUILAR	203-33-3234	40,000
BENSON	323-22-2943	60,000
D'SOUZA	989-22-2403	80,000
FIORANO	093-44-3823	45,000



LAST_NAME	SSN	SALARY
ANSKEKSL	111–23-1111	40,000
BKJHHEIEDK	111-34-1345	60,000
KDDEHLHESA	111-97-2749	80,000
FPENZXIEK	111-49-3849	45,000

Figure 3: Sample data masking. After masking, true content and meaning of data shouldn't be seen from it, and their own relationship shouldn't be destroyed.

The process of data masking as shown below, you need to first cloned production data to a temporary Library in the middle, and then perform a data mask in this middle library, through the shielding the database can be used as a test library, also can be cloned into other test environments.

Note that the data masking is irreversible.

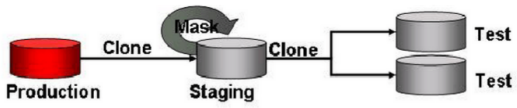


Figure 4: Sample data masking process.

Oracle's DataMasking (Pack) can mask sensitive data in the database by formatting the values of a column in the specified table. Data masking includes: shuffle, random numbers, random characters, columns of a reference table, etc.

4. Implementation of two tools

4.1. Implementation of HTML tool

It's run in Windows System which requires python 2.7 and scrapy framework. Since it's must in a Internet-disconnected machine, library can be installed in a machine on the network, and then copy python27 to it.
For html parsing and XML parsing, users input read path and write path, then the program will give the converted program. The format should be stored in some .csv file.
The general idea of this algorithm is as follow:

Algorithm 1 Crawling Algorithm for callLog

```
1: procedure MyProcedure
2: initial:
3:     url ← find global address of content0.html
4:     response ← scrapyRequest(url)
5: selection:
6:     calladdress ← select address for callLog
7:     cut the key segments base on the key words
8:     terminal pages ← key segments cut
9: parsing:
10:    establish .csv file in specific output path
11:    open .csv file in specific output path
12:    select segment to write according to its format
13: end procedure
```

It's just one of several algorithms, the information about message and address book is also needed, but the procedure of crawling them is similar. What's different between them is the call log consider MD5 additionally which occurs in .csv files. I'll illustrate it, first we should read the input path and open the start HTML. Then for every HTML file we use HtmlResponse for getting contents of webpages. We observe the characteristics of pages, pick it out t from HTML. For example, we should first find the sentence start with "\\td", then extract with "./span/img", by marking location of CallRecords.ico, we can find the next pages we want to search, after some similar procedures, we can get the address of next html, so we continue to crawling it.

4.2. Implementation of XML tool

The crawling of XML files is different from HTML. The content of HTML is shown in different html files. For XML, it's shown in different .bcp file, it just like excel file, and some transformation problems are also involved. There exists problems that if the path use Chinese path, it may causes error. To avoid it, an additional information would tell users that they should use correct English path.

5. Implementation of web framework

In this module, I cooperate with Yaowei Huang and Zhige Li, we use Django framework as our model. Some softwares should be installed, which are *python-2.7.11* and *Djang-1.6.11*. After setting up environment, only use several commands to start a new project which has the following structure:

[*manage.py*] a utility command line tool that lets you interact with the Django project in a variety of ways.

[*__init__.py*] an empty file, and tell Python that the directory is a Python package.

[*settings.py*] the settings / configuration of the Django project.

[*urls.py*] the URL statement for the Django project; a web directory driven by Django".

[*wsgi.py*] the entrance of a Web server which is compatible with WSGI for running your project.

[*manage.py*] an app, after setting it, new name should be added into setting.py. It should be started with a specified command. Also, you should start a new file called *urls.py*, which belongs to urls.py of the root content.

Next, we should establish template file and add all *html* file into it with configuration of *setting.py*. For *html* file, focus on configuration of *urls.py* and *views.py*, the former manages *view.py*, the latter is responsible

for implement of function of websites. What we've done is designing two pages, one for sign up, one for uploading files, another for show the result, success or fail.

6. data masking

First, from now on, our work has no relationship with the former work, It is another work and I'm still work for it with two other classmates.
Since the data is secret, so in case that our work leakages the information of criminal, we should mask data.
For simple and understandable, here is a simple describe for it.

Example: ID number 51072219760927297X.

For 1 to 6 bits, use an alphabetic sequence and use numbers as offsets,remaining bit is converted into a sixteen bit bit-string and then turns it into an octal digit, which is 510722, use HFDJKA(save by the data provider to recover)

H+5=M, F+1=G, D+0=D, J+7=Q, K+2=M, A+2=C
At present, it has changed into MGDQMC.
For remaining number,19760927297X, according to the idea, it should converted into:

0001,1001,0111,0100,
0000,1001,0010,0111,
0010,1001,0111,1111

Then reorganize it by 3bit string:

000,110,010,111,
010,000,001,001,
001,001,110,010,
100,101,111,111

Recalculate by 3 bits, finally we have:

0627201111624577

The ID card is finally become:

MGDQMGC0627201111624577

you need to save MGDQMC. For other transaction information, this method is also available. It's just a simple version of encryption, In practice, there exists some other additional algorithm to avoid them being recognized. We're rewriting the code for more complex encryption.

7. Visualization

For encrypted data, we use several algorithm to cluster them. For clustering, there are many famous algorithm, we mainly use PRA algorithm.

The Path Ranking Algorithm (PRA) [25, 26] extends the idea of using random walks of bounded lengths for predicting links in multi-relational knowledge graphs. In particular, let$\pi_L(i,j,k,t)$ denote a path of length L of the form $e_i \xrightarrow{r_1} e_2 \xrightarrow{r_2} e_3 ... \xrightarrow{r_L} e_j$, where t represents the sequence of edge types $t = (r_1, r_2, ..., r_L)$. We also require there to be a direct arc $e_i \xrightarrow{r_1} e_j$, representing the existence of a relationship of type k from e_i to e_j Let $\Pi_L(i,j,k)$ represent the set of all such paths of length L,ranging over path types t. (We can discover such paths by enumerating all (type-consistent) paths from entities of type e_i to entities of type e_j . If there are too many relations to make this feasible, we can perform random sampling.) We can compute the probability of following such a path by assuming that at each step, we follow an outgoing link uniformly at random. Let $P(\pi_L(i,j,k,t))$ be the probability of this particular path; this can be computed recursively by a sampling procedure, similar to PageRank (see [26] for details). The key idea in PRA is to use these path probabilities as features for predicting the probability of missing edges. More precisely, define the feature vector:

$$\phi_{ijk}^{PRA} = [P(\pi) : \pi \in \Pi_L(i,j,k)]$$

We can then predict the edge probabilities using logistic regression:

$$f_{ijk}^{PRA} := w_k^T \phi_{ijk}^{PRA}$$

Interpretability: A useful property of PRA is that its model is easily interpretable. In particular, relation paths can be regarded as bodies of weighted rules ł more precisely Horn clauses ł where the weight specifies how predictive the body of the rule is for the head. We've accomplished the visualization. Here is a simple graph.
Through clicking the nodes and edges, we can know the

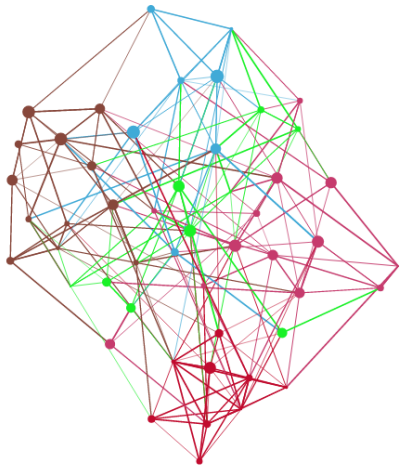


Figure 5: A Graph for Criminal

information of criminal. Police can check their relation and information conveniently. Besides, we use Chinese word segmentation tool to understand content of users, so they can return a correct result what they want.

8. Future work

Based on what we've done, I'll explain it separately:
i).Data masking is strict, it's sensitive for police to send information out of police. So they want us to encrypt data again in more complex way, such as RSA algorithm. In addition, they have some divergence on specified segments. Since our method expands the length of original string, so we should encrypt it with same length.
ii).Visualization is accomplished, the functions of searching haven't finished yet. We also need to make some link prediction for key points, at the same time I also need to get more data from police. That's what we would do next.

9. Conclusion

In this program, I made two tools for police and start a web-framework based on python Django, then our work turns to another direction which aims for establishing a graph model, so I've read many papers for knowledge, I'm really work in lab, I want to continue to work on it until the final work accomplished.

Acknowledgments

I would like to thank the help of Professor Mrs.Gan, students Zhige Li, Yaowei Huang, Fengyu Deng and Weitang Chen. They give me lots of idea and suggestion about the subject, and they also take part in the project. Thanks again!

References

[1] R. Davis, H. Shrobe, and P. Szolovits, "What is a knowledge representation?" AI Magazine, vol. 14, no. 1, pp. 17C33, 1993.

[2] J. F. Sowa, "Semantic networks," Encyclopedia of Cognitive Science, 2006.

[3] M. Minsky, "A framework for representing knowledge," MIT-AI Laboratory Memo 306, 1974.

[4] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," 2001. [Online]. Available: http://www.scientificamerican.com/article/the-semantic-web/

[5] T. Berners-Lee, "Linked Data - Design Issues," Jul. 2006. [Online]. Available: http://www.w3.org/DesignIssues/LinkedData.html

[6] C. Bizer, T. Heath, and T. Berners-Lee, "Linked data-the story so far," International Journal on Semantic Web and Information Systems, vol. 5, no. 3, pp. 1C22, 2009.

[7] G. Klyne and J. J. Carroll, "Resource Description Framework (RDF): Concepts and Abstract Syntax," Feb. 2004. [Online]. Available: http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/

[8] R. Cyganiak, D. Wood, and M. Lanthaler, "RDF 1.1 Concepts and Abstract Syntax," Feb. 2014. [Online]. Available: http://www.w3.org/TR/2014/RECrdf11-concepts-20140225/

[9] R. Brachman and H. Levesque, Knowledge Representation and Reasoning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2004.

[10] J. F. Sowa, Knowledge Representation: Logical, Philosophical and Computational Foundations. Pacific Grove, CA, USA: Brooks/Cole Publishing Co., 2000.

[11] G. Angeli and C. Manning, "Philosophers are Mortal: Inferring the Truth of Unseen Facts," in Proceedings of the Seventeenth Conference on Computational Natural Language Learning. Sofia, Bulgaria: Association for Computational Linguistics, Aug. 2013, pp. 133C142.

[12] B. Taskar, M.-F. Wong, P. Abbeel, and D. Koller, "Link Prediction in Relational Data," in Advances in Neural Information Processing Systems, S. Thrun, L. Saul, and B. Scholkopf, Eds., vol. 16. Cambridge, MA: MIT Press, 2004.

[13] L. Getoor and C. P. Diehl, "Link mining: a survey," ACM SIGKDD Explorations Newsletter, vol. 7, no. 2, pp. 3C12, 2005.

[14] H. B. Newcombe, J. M. Kennedy, S. J. Axford, and A. P. James, "Automatic Linkage of Vital Records Computers can be used to extract "follow-up" statistics of families from files of routine records," Science, vol. 130, no. 3381, pp. 954C959, Oct. 1959.

[15] S. Tejada, C. A. Knoblock, and S. Minton, "Learning object identification rules for information integration," Information Systems, vol. 26, no. 8, pp. 607C633, 2001.

[16] E. Rahm and P. A. Bernstein, "A survey of approaches to automatic schema matching," the VLDB Journal, vol. 10, no. 4, pp. 334C350, 2001.

[17] A. Culotta and A. McCallum, "Joint deduplication of multiple record types in relational data," in Proceedings of the 14th ACM international conference on Information and knowledge management. ACM, 2005, pp. 257C258.

[18] P. Singla and P. Domingos, "Entity Resolution with Markov Logic," in Data Mining, 2006. ICDM '06. Sixth International Conference on, Dec. 2006, pp. 572C582.

[19] I. Bhattacharya and L. Getoor, ”Collective entity resolution in relational data,” ACM Trans. Knowl. Discov. Data, vol. 1, no. 1, Mar. 2007.

[20] S. E. Whang and H. Garcia-Molina, ”Joint Entity Resolution,” in 2012 IEEE 28th International Conference on Data Engineering. Washington, DC, USA: IEEE Computer Society, 2012, pp. 294C305.

[21] S. Fortunato, ”Community detection in graphs,” Physics Reports, vol. 486, no. 3, pp. 75C174, 2010.

[22] R. West, E. Gabrilovich, K. Murphy, S. Sun, R. Gupta, and D. Lin, ”Knowledge Base Completion via Search-Based Question Answering,” in Proceedings of the 23rd International Conference on World Wide Web, 2014, pp. 515C526.

[23] G. Weikum and M. Theobald, ”From information to knowledge: harvesting entities and relationships from Web sources,” in Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. ACM, 2010, pp. 65C76.

[24] B. Suh, G. Convertino, E. H. Chi, and P. Pirolli, ”The Singularity is Not Near: Slowing Growth of Wikipedia,” in Proceedings of the 5th International Symposium on Wikis and Open Collaboration. New York, NY, USA: ACM, 2009, pp. 8:1C8:10.

[25] N. Lao and W. W. Cohen, ”Relational retrieval using a combination of path-constrained random walks,” Machine learning, vol. 81, no. 1, pp. 53C67, 2010.

[26] N. Lao, T. Mitchell, and W. W. Cohen, Random walk inference and learning in a large scale knowledge base, in Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011, pp. 529C539.