|  |  |
| --- | --- |
| Name: | Wu Jian |
| Supervisor(s): | Professor Ben C.M. Kao |
| Dissertation Title: | Machine Comprehension of Court Cases |
| Planned Submission Semester: |  |

**Aim**

|  |
| --- |
| The purpose of the dissertation is to develop a Knowledge graph on an existing law dataset to assist judges in adjudicating cases. The Knowledge graph is try to Captures entities, attributes, and relationships, which can help the judge to do the QA and decision support.  The main mission of this dissertation is to generate structured query based on the judge’s question and build a knowledge graph. |

**Brief Literature Review**

|  |
| --- |
| Question-Answering Module:  The QA task is to extract the question into structured query by using the NER and relation extraction.   1. Named Entity Reorganization:   we choose linguistic grammar-based techniques to train the model, and the most common model is LSTM-CRF networks. This network can efficiently use past input features via a LSTM layer and sentence level tag information via a CRF layer. With such a CRF layer, we can efficiently use past and future tags to predict the current tag.[1][2]   1. Relation extraction:   Multi-channel convolutional neural networks (CNNs) are utilized to determine the relations between a pair of entities in a given free question (Xu, et al., 2016). Specifically, two CNNs channels are used. One is used to capture syntax information and the other is to capture context information. The convolutional layer of each channel accepts an input of variable length, while returns a vector of fixed length using the Maximum Sampling method. These fixed-length vectors are combined together to form the input of final softmax classifier, whose output vector dimension equals to the total number of relation categories and the value of each dimension equals to the degree of confidence mapped to the corresponding predicates in the knowledge graph.[3]  Knowledge Graph Module:  There are two types of the approaches in the development of KGs: top-down approach focusing on knowledge schema such as the domain ontologies and bottom-up approach focusing on knowledge instances such as Linked Open Data (LOD) datasets. In this dissertation, we want to build a bottom-up knowledge graph.  The general procedures of KG development consist of three phases: knowledge extraction, knowledge fusion and knowledge inference.  Knowledge extraction consists of entity extraction, attribute extraction and relation extraction.  Entity extraction, including named-entity recognition (NER), is to discover entities from a wide variety of knowledge resource and try to classify them into pre-defifined categories such as person, location, organization, news title, service,time, date, and so on.  Relation extraction needs to extract the semantic relationship between two or more entities from the text. The main methods are as follows:**hand-written patterns, supervised machine learning, semi-supervised and unsupervised.**[4]  After the knowledge extraction, we should do the knowledge fusion which consists coreference resolution and entity disambiguation. Through information extraction, we can get the attribute information of entities, relationships and entities from the original unstructured and semi-structured data.  If we compare the following process to a jigsaw puzzle, then the information is a jigsaw puzzle fragments, scattered, and even fragments from other jigsaw puzzles, which themselves are the wrong pieces to interfere with our jigsaw puzzle.  Finally, we also need to do the quality evaluation and ontology construction. |
| [1] Han, Li-Feng Aaron, Wong, Fai, Chao, Lidia Sam. (2013). Chinese Named Entity Recognition with Conditional Random Fields in the Light of Chinese Characteristics. Proceeding of International Conference of Language Processing and Intelligent Information Systems. M.A. Klopotek et al. (Eds.): IIS 2013, LNCS Vol. 7912, pp. 57–68 [[1]](https://link.springer.com/chapter/10.1007%2F978-3-642-38634-3_8#page-1)  [2] Jenny Rose Finkel; Trond Grenager; Christopher Manning (2005). Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling (PDF). 43rd Annual Meeting of the Association for Computational Linguistics. pp. 363–370.  [3]Ziqing Liu , Enwei Peng , Shixing Yan , Guozheng Li, Tianyong Hao (2018) T-Know: A Knowledge Graph-based Question Answering and Information Retrieval System for Traditional Chinese Medicine(PDF).  [4]Zhanfang Zhao,Sung-Kook Han,In-Mi So(2018). [Architecture of Knowledge Graph Construction Techniques(PDF).](https://acadpubl.eu/jsi/2018-118-19/articles/19b/24.pdf) |

**Proposed Methodology**

|  |
| --- |
| 1. We using BiLSTM + CRF+ CNN to do the NER and relation extraction in both question-answering module and the knowledge graph module. 2. Entity disambiguation: we use the VSM(vector space model) to transform entities into low dimensional and dense vectors. We calculate the distance of these vectors by computing the cosine similarities of them. 3. Coreference resolution: Try the mention pair model to train a binary classifier that assigns every pair of mentions a probability of being coreferent. e.g., for “he” look at all candidate antecedents (previously occurring mentions) and decide which are coreferent with it. 4. Using the neo4j to store the knowledge graph. |

**Milestones**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Tasks*** | | ***Estimated completion time*** | ***Estimated number of***  ***learning hours*** |
| 1 | Build the BiLSTM + CRF+ CNN to do the NER and relation extraction | Before 31.8.2019 | 120 hours |
| 2 | Train and tune the model | Before 9.31.2019 | 120 hours |
| 3 | Finish the entity disambiguation | Before 31.10.2019 | 120 hours |
| 4 | Build the Coreference resolution | Before 31.11.2019 | 120hours |
| 5 | Build the neo4j environment and construct the knowledge graph | Before 31.12.2019 | 120hours |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |
|  |  |  | ***Total: 600*** |

**Deliverables**

|  |  |
| --- | --- |
| ***Items*** | |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |