# Instructions for the SI630 Project Proposal

**Version 1.0**

**Jiale Zha** `jialezha@umich.edu`


## Abstract

Relation Extraction, as an important part of Information Extraction, has always attracted our attention. Until now, various methods have been utilized in this area, including feature-based method, kernel-based method, nueral network method and so on. Our project will focus on those popular models for relation extraction system, try to compare the performance of them and design a new model combining their advantages.


## 1 Introduction

Extracting information from textual data has always been a popular topic in natural language processing(NLP) area. Of all those information, the relationship information always attracts our attention first. Thus, Relation Extraction(RE) plays a vital role in this area. Basically, relation extraction is to extract semantic relations, like belonging to, between eneities, such as person, organization, location and so on. Table 1 gives a example of relationship extraction, the entity pair ***henrik stenson*** and ***sweden*** has the relationship ***country of citizenship***.

In some situation, RE system even could provide us with the majority information of text, so it's widely used for further textual learning problem, like the construction of knowledge graph (Milosevic and Thielemann (2022)), the exploration of gene-disease relationships (Chun et al. (2006)), protein-protein interaction (Huang et al. (2004)) and so on.

Having known that relation extraction is crucial to many further problems, in our project, we want to keep track of those methods used in this area, especially feature-based, kernel-based and deep learning methods, all of which we will describe with details in the following section, and to see whether they could provide us with some idea to make a new prograss.

## 2 Problem Definition

The framework for relation extraction typically could be divided into two steps (Nguyen and Grishman (2015)): 1. for some given entity pairs of interest, we want to detect whether a relation utterance corresponding to them in a sentence represents some relations; 2. if some relationships are detected, we want to classify them into some predefined classes, which is also known as *Relation Classification(RC)*.

One feature of this problem is that the training dataset will contains much more examples without any relation than those with relation, so it will be tremendously unbalanced and we could not just treat those without relation as one class. This feature is also the reason why RE system is more challenging but practical than simple RC system.

The main idea for solving this problem has been changed over time. At the very beginning, researchers tend to find a rich structural representation of raw text data(Bach and Badaskar (2007)), and then use binary classifier to detect whether a particular kind of relationship exists. Featured-based and kernel-based methods(Bach and Badaskar (2007)) are two main ideas to capture the structure of a sentence, and the features constructured by them are also known as hand-designed features. Those features that are most popular include (Bach and Badaskar (2007)) (1) the entities themselves, (2) the types of the two entities, (3) word sequence between the entities, (4) number of words between the entities and (5) path in the parse tree containing the two entities.

Those traditional methods could utilize the structure of text data, but the wrong representation of it could also bring errors to the final classification. So in the last decade, as the development of neural network(NN), especially those designed for NLP tasks, such as RNN and LSTM, a new kind

| **Sentence**: "<e1>Henrik Stenson </e1> of < e2> Sweden </e2> will replace Harrington in the field." | | |
|---|---|---|
| **Entity 1**: *henrik stenson* | **Entity 2**: *sweden* | **Relation**: *country of citizenship* |

Table 1: A Example of Relationship Extraction

of end-to-end model appears. This kind of model will directly take the sentence and the position of two entities as input, the neural network will automatically capture the intrinsic structures and use the multi-class classifier like softmax to give prediction.

In our project, we want to compare the performance of those traditional methods with hand-designed features with that of those end-to-end models. And we will try to combine these two kinds of approaches together, that is to use the information from those hand-designed features in neural network.

## 3 Related Work

The paper given by Bach and Badaskar (2007) provide a review of those tradition methods, including a brief introduction to some feature-based methods(Kambhatla (2004)) and kernel-based methods(Zhao and Grishman (2005)). Featured-based methods focus more on those statistics about the entities and context, while kernel-based methods pay more attention to the useful transformation of original text, and so they are more flexible. Some classic kernels includes tree kernel(Zelenko et al. (2003)), subsequence kernel(Bunescu and Mooney (2005b)), dependency tree kernel(Bunescu and Mooney (2005a)) etc.

As we mentioned before, the new end-to-end model is brought by neural network framework, and it is the rise of *convolutional neural network(CNN)* give this framework a second life. So this kind of neural network is first introduced in to the area of RE system, in Nguyen and Grishman (2015). After that, some neural networks specially designed for NLP appear and are used for RE system, for example, recursive neural network(RNN) in Socher et al. (2012) and Long Short-Term Memory(LSTM) in Miwa and Bansal (2016).

## 4 Data

Basically, since relation extraction is a very hot topic for NLP, a lot of famous dataset could be used to train and test our models, such as NYT (Riedel et al. (2013)), ACE2005 (Nguyen and Grishman (2015)), Wiki (Han et al. (2020)) and so on. These datasets, however, have been explored by many researchers, so we decide to construct a new dataset from them.

In our final project, we will use NYT10, NYT10m, Wiki20m and Wiki80 as our original dataset, since they have very similar form and labels. The following figures 1 and 2 provide two examples from them.

| text | He is a son of Vera and William Lichtenberg of Belle Harbor, Queens. |
|---|---|
| h | id: m.0ccvx name: Queens pos: [62, 68] |
| t | id: m.05gf08 name: Belle Harbor pos: [47, 59] |
| relation | /location/location/contains |

Figure 1: An example of NYT dataset

| token | "The", "Paris", "Bourse", ",", "now", "part", "of", "Euronext", ",", "an", "order", "-", "driven", ",", "electronic", "stock", "exchange", "." |
|---|---|
| h | id: Q2385849 name: paris bourse pos: [1, 3] |
| t | id: Q11691 name: stock exchange pos: [16, 18] |
| relation | instance of |

Figure 2: An example of Wiki dataset

In our project, we will only remain the information of original text, the name of entities and their relation, and use these to construct the input data frame. The figure 3 below show a row of that input data frame.

What's more, the representation of labels in two datasets are different, so we have to combine them together. In our project, we will remain the labels of Wiki dataset and change those in NYT dataset. The map we use to justify labels in NYT dataset is shown in Appendix A

| text | The Paris Bourse, now part of Euronext, an order-driven, electronic stock exchange. |
|---|---|
| entity one | paris brouse |
| entity two | stock exchange |
| relation | instance of |

Figure 3: An example of data in our project

# 5 Methodology

As what we have mentioned before, our project will keep track of those tradition and fancy methods used for RE system, so basically, we will first train three kinds of model: feature-based, kernel-based and deep learning model.

## 5.1 Feature-Baed Model

This kind of model is a classificer model based on hand-designed features, including

1. the entities themselves
2. the types of the two entities
3. word sequence between the entities
4. number of words between the entities
5. path in the parse tree and dependency tree

Since all of the above features could be derived from syntactic parse tree and dependency tree (Kambhatla (2004)), we then will utilize some NLP toolkits, like $nltk$, to build those two trees for each piece of data and ocnstruct those features. Finally, we will use the softmax classifier to extract the relation.

## 5.2 Kernel-Based Model

The kernel-based model is named after the kernel method used in the famous classifier, $Support\ Vector\ Machine$(SVM). For RE system, it utilizes the string-kernels (Lodhi et al. (2002)), which is based on the context subsequence of entites in the text, and the most popular kernels used are $bag\ of\ features\ kernel$ (Bunescu and Mooney (2005a)) and $tree\ kernel$ (Zelenko et al. (2003)).

In our project, we will follow the two ideas above, compute two kernel functions for each piece of data, and then train the corresponding SVM classifiers.

## 5.3 Deep Learning Model

A deep learning model, which is end-to-end, will take the text and the position of entities(or relative position of context) as input, pass them through a network framework, and then use the softmax classifier to give a prediction.

In our project, we will utilize the very famous network in NLP field, the Bi-Directional LSTM model.

## 5.4 Combined Model

After training the above three kinds of model, we will try to design a combined model of them, which will use the neural network as main framework and will take those hand-designed features as additional input.

Ideally, since we provide more direct information here, this combined model should work better than the previous three.

# 6 Evaluation and Results

For RE system, since our final target will always become a classificatoin for several pre-defined relation labels, it's typical to use those metrics like accuracy precision, recall and F1 score to evaluation models. And according to our target, to compare the preformance of traditional methods with new end-to-end model and to combine them together, we could just use the traditional method as our baseline, so that we could take a look at whether those variation will improve the traditional model or not.

# 7 Work Plan

To achieve our final target, we have to do some preparation: to understand the framework of both traditional methods and neural network models and to implement the above two kinds of models by ourselves. Then we could try to design a model which combines them together and do an experiment to compare all of them.

So our planned timeline is as follows. By now, we have read several paper about relation extraction in the past weeks, and built an basic bi-directional LSTM based on pytorch.

In the following several weeks, we will try to implement a feature-based and a kernal-based model. Finally, based on those previous work, we will design our own neural network model with hand-designed features as part of input information, and do some experiment to see its performance.

# References


Nguyen Bach and Sameer Badaskar. 2007. A review of relation extraction. *Literature review for Language and Statistics II* 2:1–15.

Razvan Bunescu and Raymond Mooney. 2005a. A shortest path dependency kernel for relation extraction. In *Proceedings of Human Language Technol-*

ogy Conference and Conference on Empirical Methods in Natural Language Processing*. pages 724–731.

Razvan Bunescu and Raymond J Mooney. 2005b. Subsequence kernels for relation extraction. In *NIPS*. Citeseer, pages 171–178.

Hong-Woo Chun, Yoshimasa Tsuruoka, Jin-Dong Kim, Rie Shiba, Naoki Nagata, Teruyoshi Hishiki, and Jun'ichi Tsujii. 2006. Extraction of gene-disease relations from medline using domain dictionaries and machine learning. In *Biocomputing 2006*, World Scientific, pages 4–15.

Xu Han, Tianyu Gao, Yankai Lin, Hao Peng, Yaoliang Yang, Chaojun Xiao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2020. More data, more relations, more context and more openness: A review and outlook for relation extraction. *arXiv preprint arXiv:2004.03186* .

Minlie Huang, Xiaoyan Zhu, Yu Hao, Donald G Payan, Kunbin Qu, and Ming Li. 2004. Discovering patterns to extract protein–protein interactions from full texts. *Bioinformatics* 20(18):3604–3612.

Nanda Kambhatla. 2004. Combining lexical, syntactic, and semantic features with maximum entropy models for information extraction. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*. pages 178–181.

Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini, and Chris Watkins. 2002. Text classification using string kernels. *Journal of machine learning research* 2(Feb):419–444.

Nikola Milosevic and Wolfgang Thielemann. 2022. Relationship extraction for knowledge graph creation from biomedical literature. *arXiv preprint arXiv:2201.01647* .

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. *arXiv preprint arXiv:1601.00770* .

Thien Huu Nguyen and Ralph Grishman. 2015. Relation extraction: Perspective from convolutional neural networks. In *Proceedings of the 1st workshop on vector space modeling for natural language processing*. pages 39–48.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. pages 74–84.

Richard Socher, Brody Huval, Christopher D Manning, and Andrew Y Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*. pages 1201–1211.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *Journal of machine learning research* 3(Feb):1083–1106.

Shubin Zhao and Ralph Grishman. 2005. Extracting relations with integrated information using kernel methods. In *Proceedings of the 43rd annual meeting of the association for computational linguistics (acl'05)*. pages 419–426.

## A Label map used in NYT

The label map for labels in NYT dataset is shown in the following two pages.

| Original Label | New Label |
|---|---|
| /location/location/contains | contains administrative territorial entity |
| /people/person/nationality | country of citizenship |
| /people/person/place_of_birth | place of birth |
| /people/deceased_person/place_of_death | place of death |
| /people/person/place_lived | place live in |
| /business/company/founders | founded by |
| /people/person/ethnicity | ethnicity of |
| /location/neighborhood/neighborhood_of | neighorhood of |
| /business/person/company | works for company |
| /location/administrative_division/country | located in the administrative territorial entity |
| /business/company/place_founded | place founded |
| /location/country/administrative_divisions | has administrative territorial entity |
| /location/country/capital | has capital |
| /people/person/children | child |
| /sports/sports_team_location/teams | has sports team |
| /people/person/religion | religion |
| /people/ethnicity/people | person of ethnicity |
| /business/company/major_shareholders | has major shareholders |
| /business/company_shareholder/major_shareholder_of | major shareholder of |
| /people/ethnicity/geographic_distribution | geographic distribution |
| /business/company/industry | industry |
| /sports/sports_team/location | location of sports team |
| /people/person/profession | profession of |
| /business/company/advisors | has advisor |
| NA | NA |
| /location/us_county/county_seat | county seating at |
| /film/film/featured_film_locations | has featured location |
| /people/ethnicity/included_in_group | included in group |
| /people/place_of_interment/interred_here | buries him/or |
| /time/event/locations | location |
| /location/de_state/capital | 'captial of german state' |
| /location/us_state/capital | captial of us state |
| /business/company_advisor/companies_advised | advises company |
| /people/deceased_person/place_of_burial | place of burial |
| /broadcast/content/location | broadcast of |

Table 2: Label Map for NYT (1)

| Original Label | New Label |
| --- | --- |
| /film/film_festival/location | film festival at |
| /location/it_region/capital | captial of it region |
| /business/shopping_center_owner/shopping_centers_owned | own shopping center |
| /location/in_state/legislative_capital | legislative captial of in state |
| /location/in_state/administrative_capital | administrative captial of in state |
| /business/business_location/parent_company | headquarter location for |
| /people/family/members | family of |
| /location/jp_prefecture/capital | captial of jp prefecture |
| /film/film_location/featured_in_films | featured location for film |
| /people/family/country | family at |
| /business/company/locations | headquarter locates at |
| /people/ethnicity/includes_groups | includes group |
| /people/profession/people_with_this_profession | people with profession |
| /location/br_state/capital | captial of br state |
| /location/cn_province/capital | captial of cn state |
| /broadcast/producer/location | broadcast at |
| /location/fr_region/capital | captial of fr state |
| /location/province/capital | captial of province |
| /location/in_state/judicial_capital | judicial captial of in state |
| /business/shopping_center/owner | owned by |
| /location/mx_state/capital | captial of mx state |
| /people/deceasedperson/place_of_death | place of death |
| /business/company/majorshareholders | has major share holder |
| /business/location | company location |
| /location/region/capital | capital of region |
| /people/deceasedperson/place_of_burial | place of burial |

Table 3: Label Map for NYT (2)