Hierarchical Relation Extraction with Encoder-Decoder model

Yufan Cai
Shanghai Jiao Tong University

1. INTRODUCTION

There is a big problem in relation extraction that building a machine learning system needs a lot of training examples. One common technique for coping with this difficulty is distant supervision (Mintz et al., 2009) which assumes that if two entities have a relationship in a known knowledge base, then all sentences that mention these two entities will express that relationship in some way. Take an example to show the automatic labeling of data through distant supervision in Figure 1. Alibaba and Jack Ma are two related entities. All sentences that contain these two entities are selected as training instances. The distant supervision techniques works effectively in automatically labeling training data, but it has two major weakness when used for relation extraction.

First, the assumption of the distant supervision is too strong to get the true label. Sentences that mentions two entities does not necessarily express the same relation in a knowledge base. It is possible that these two entities may simply share the same topic. For instance, the sentence 1 and sentence 2 in the Figure 1 shows the relation ”business/company/founders” relation while the sentence 3 does not express this relation but is still selected as a training instance. This sentence plays a role of noisy data that hinder the performance of the model.

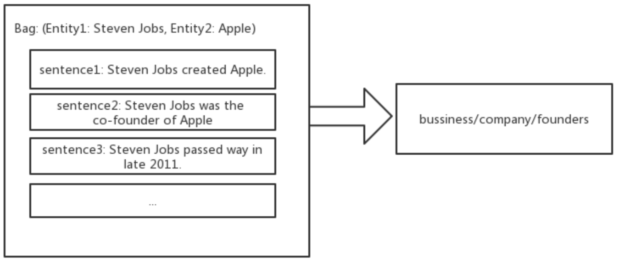
Bag: (Entity1: Steven Jobs, Entity2: Apple)
sentence1: Steven Jobs created Apple.
sentence2: Steven Jobs was the co-founder of Apple
sentence3: Steven Jobs passed way in late 2011.
...
bussiness/company/founders

Fig. 1. Training instances generated through distant supervision. Sentence 1 and 2: correct labeling; Sentence 3: incorrect labeling.

Second, previous methods(Mintz et al., 2009;Riedel et al., 2010; Hoffmann et al., 2011) have typically applied supervised models to elaborately designed features when obtained the labeled data through distant supervision. These features are often derived from preexisting Natural Language Processing (NLP) tools. Since errors inevitably exist in NLP tools, the use of traditional features leads to error propagation or accumulation. Meanwhile, distant supervised relation extraction generally addresses corpora from the Web, including many informal texts. McDonald and Nivre(2007) showed that the accuracy of syntactic parsing decreases significantly with increasing sentence length. Therefore, when using traditional features, the problem of error propagation or accumulation will not only exist, it will grow more serious.

In this paper, we propose a novel model called Hierarchical Relation Extraction with Encoder-Decoder model with multi-instance learning to address the two problems described above. To address the first problem, distant supervised relation extraction is treated as a multi-instance problem similar to previous studies (Riedel et al., 2010; Hoffmann et al.,2011; Surdeanu et al., 2012). In multi-instance problem, the training set consists of many bags, and each contains many instances. The labels of the bags are known; however, the labels of the instances in the bags are unknown. We design an objective function at the bag level. In the learning process, the uncertainty of instance labels can be taken into account; this alleviates the wrong label problem.What’s more, we adopt hierarchical output which activated from the hierarchical clustering. We know the relation in large data set is also hierarchical organized, so it’s nature to output the relation step by step. We first confirm the main domain of the relation and then elaborate the relation in smaller domain. This method will help to pinpoint the relation and make it more accuracy.

Based on the above observation, in this report, we present an encoder-decoder model for distant supervised relation extraction. Specifically, given an entity pair and its sentence bag as input, in the encoder component of our model, we employ the BGRU neural network to extract the features of the sentences in the sentence bag and merge them into a bag presentation. In the decoder component of our model, we utilize the long short-term memory network to model relation dependencies and predict the target relations in a sequential manner. In this sequential procedure, the relations should be predicted in a way that, the relations having more information in the bag representation are predicted earlier and used as prior knowledge for further predictions.

Additionally, we incorporate the attention mechanism into our model which dynamically adjusts the bag representation to reduce the impact of sentences whose corresponding relations have been predicted. We conduct extensive experiments on a widely used dataset released by [Riedel et al., 2010]. Experimental results show that our model significantly and consistently outperforms state-of-the-art methods.

To address the second problem, we adopt Bidirectional Gated Recurrent Unit architecture to automatically learn relevant features without complicated prepossessing. There are many effective way in the past. For example, in Zeng et al. (2014), they used a single max pooling operation utilized to determine the most significant features. Although this operation has been shown to be effective for textual feature representation (Collobert et al., 2011; Kim, 2014), it reduces the size of the hidden layers too rapidly and cannot capture the structural information between two entities (Graham, 2014). For example, to identify the relation between Steve Jobs and Apple in Figure 1, we need to specify the entities and extract the structural features between them.

Several approaches have employed manually crafted features that attempt to model such structural information. These approaches usually consider both internal and external contexts. A sentence is inherently divided into three segments according to the two given entities.The internal context includes the characters inside the two entities, and the external context involves the characters around the two entities (Zhang et al., 2006).

To capture structural and other latent information, we divide the feature input vectors into three segments based on the positions of the two given entities and devise a piece-wise soft-max layer instead of the single soft-max layer. The piece-wise soft-max layer procedure returns the maximum value in each segment instead of a single maximum value over the entire sentence. Thus, it is expected to exhibit superior performance compared with traditional methods.

The contributions of this report can be summarized as follows:

- We explore the feasibility of performing distant supervised relation extraction using two BGRU network with selective attention model.
- To address the wrong label problem, we develop innovative solutions that using hierarchical techniques and the position vectors of the two entities.

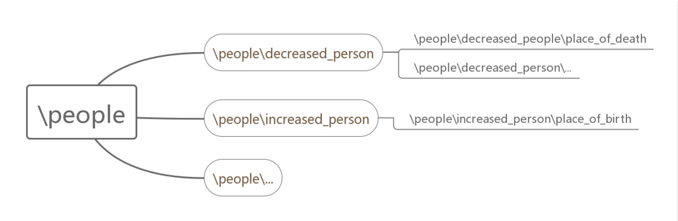
\people
\people\decreased_person
\people\decreased_people\place_of_death
\people\decreased_person\...
\people\increased_person
\people\increased_person\place_of_birth
\people\...

Fig. 2. Hierarchical relation example

2. RELATED WORK IN RELATION EXTRACTION

Relation extraction is one of the most important topics in NLP. Many approaches to relation extraction have been developed, such as bootstrapping, unsupervised relation discovery and supervised classification. Supervised approaches are the most commonly used methods for relation extraction and yield relatively high performance(Bunescu and Mooney, 2006; Zelenko et al., 2003;Zhou et al., 2005). In the supervised paradigm, relation extraction is considered to be a multi-class classification problem and may suffer from a of labeled data for training.

To address this problem, Mintz et al. (2009) adopted Freebase to perform distant supervision. As described in Section 1, the algorithm for training data generation is sometimes faced with the wrong label problem. To address this shortcoming, (Riedel et al., 2010;Hoffmann et al., 2011; Surdeanu et al., 2012) developed the relaxed distant supervision assumption for multi-instance learning. The term multiinstance learning was coined by (Dietterich et al.,1997) while investigating the problem of predicting drug activity. In multi-instance learning, the uncertainty of instance labels can be taken into account. The focus of multi-instance learning is to discriminate among the bags.

These methods have been shown to be effective for relation extraction. However, their performance depends strongly on the quality of the designed features. Most existing studies have concentrated on extracting features to identify the relations between two entities. Previous methods can be generally categorized into two types:feature-based methods and kernel-based methods.

In feature-based methods, a diverse set of strategies is exploited to convert classification clues (e.g., sequences, parse trees) into feature vectors (Kambhatla, 2004; Suchanek et al., 2006). Feature-based methods suffer from the necessity of selecting a suitable feature set when converting structured representations into feature vectors. Kernel-based methods provide a natural alternative to exploit rich representations of input classification clues, such as syntactic parse trees. Kernel-based methods enable the use of a large set of features without needing to extract them explicitly.Several kernels have been proposed, such as the convolution tree kernel(Qianetal., 2008), the sub sequence kernel (Bunescu and Mooney, 2006) and the dependency tree kernel (Bunescu and Mooney,2005).

Nevertheless, as mentioned in Section 1, it is difficult to design high-quality features using existing NLP tools. With the recent revival of interest in neural networks, many researchers have investigated the possibility of using neural networks to automatically learn features (Socher et al., 2012; Zeng et al., 2014).

3. PRELIMINARIES

3.1 Task Definition

Given the training data $D = (B_i, L_i)^L$, which consists of N bags of sentences, where each bag B_i can be represented as z_i sentences such as $x_{i,1}, x_{i,2}, ..., x_{i,z_i}$. The output relations L_i is a subset of all relations$l_1, l_2, ..., l_{n_l}$, where n_l is the number of all relations. By training D, the goal of distant supervised relation extraction is to derive a proper learning model, so that the model can predict the target relations L corresponding to a given bag B.

3.2 RNN Encoder-Decoder

In this section, we briefly describe the Recurrent Neural Network Encoder-Decoder, proposed by[Sutskeveretal.,2014; Choetal., 2014], which is successfully applied to many seq2seq tasks such as machine translation [Jinchao Zhang,2017] and syntactic parsing [Vinyals et al., 2015]. In the RNN Encoder-Decoder, an encoding RNN transforms a source sequence $X = [x_1, ..., x_{T_X}]$into a fixed length vector **c**, i.e.

$$h_t = f(x_t, h_{t-1})\ c = \varphi(h_1, ..., h_{T_x}), \tag{1}$$

where h_t are the RNN hidden states, c is the context vector which is assumed as an abstract representation of X though function φ and f is a non-linear function.

Once the source sequence is encoded, another decoding RNN generates a target sequence Y = $[y_1, .., y_{T_Y}]$ through the following prediction model:

$$s_t = f(s_{t-1}, y_{t-1}, c), p(y_t|y_1, .., y_{t-1}, X) = g(s_{t-1}, y_{t-1}, c) \tag{2}$$

where s_t is RNN hidden state at time t, y_t is the predicted target symbol at time 5 t with context vector c and all the previously predicted target symbols $y_1, .., y_{t-1}$. The prediction model is typically a softmax classifier over a settled vocabulary through function g.

Attention mechanism was first introduced to RNN Encoder-Decoder[Bahdanauetal., 2014]to release the burden of summarizing the entire source into a fixed-length vector as context. The attention mechanism can dynamically choose context c t at each time step. For example, representing c t as the weighted sum of the source states h_t,

$$c_t = \sum_{\tau=1}^{T_X} \alpha_{t\tau} h_\tau;\ \alpha_{t\tau} = \frac{exp(\eta(s_{t-1}, h_\tau))}{\sum_{k=1}^{T_X} exp(\eta(s_{t-1}, h_k))}, \tag{3}$$

where η is a function to compute the attentive strength with each source hidden state, which usually adopts a multi-layer neural network.

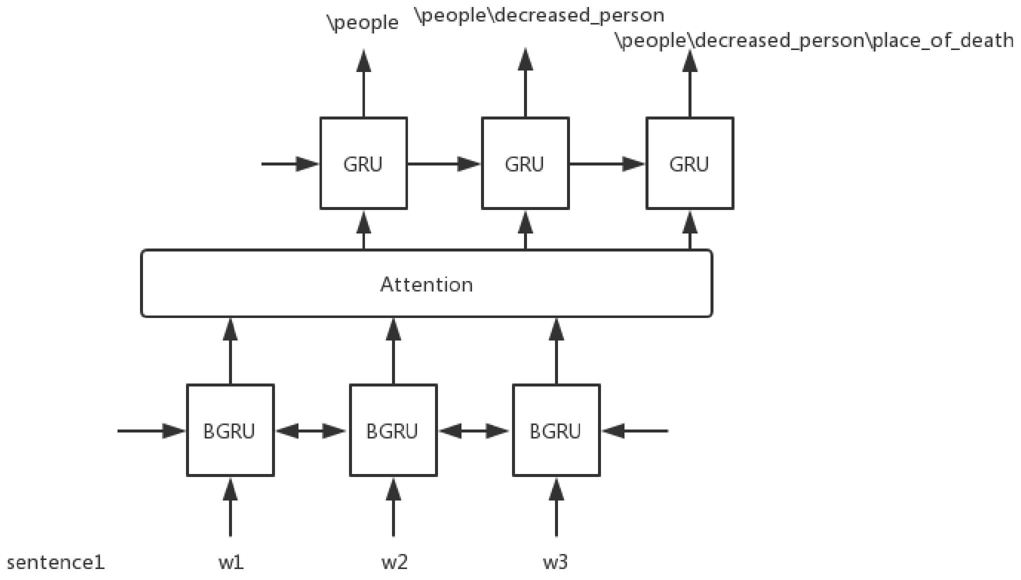

Fig. 3. The architecture of our model. In the encoder component, we employ BGRU to extract the features of each sentence, and merges them into a bag representation. In the decoder component, we utilize GRU to predict relations in a sequential manner, which directly models the dependencies among relations. The attention mechanism is incorporated into our model to dynamically adjust the bag representation.

4. THE PROPOSED MODEL

Our model consists of an encoder component and a decoder component, which takes a bag of sentences as input and gives a sequence of relations as output. The encoder component is a BGRU for capturing salient meanings of each sentence and summarizing them into a vector. Vectors of all sentences are combined into a single context vector of the whole bag, which is the input of the decoder component. The decoder component is an GRU which directly models dependencies of relations by predicting them in a sequential manner. This enables the model to use previously observed relations as prior knowledge for further predictions. The attention mechanism is additionally incorporated into our model to adjust the context vector during decoding in order to highlight sentences whose corresponding relations have not been predicted. The architecture of our model is demonstrated in Figure 2.

4.1 Encoder

Given an entity pair and its sentence bag as input, the encoder component extracts the features of sentences by CNN firstly, and then merges them into a bag representation.

$$y_i = BGRU(x_i);\ B = \eta(y_i)\ i = 1, 2, 3, ..., n \tag{4}$$

where x_i is the i-th sentence, y_i is the i-th sentence embedding obtained by BGRU, and is the function which merges the embedding of sentences into a bag representation.

4.2 Decoder

By using the encoder component, we can get the bag representation. In this section, we introduce how to model the relation dependencies and predict the target relations of the given entity pair by GRU in a sequential manner. Given a bag representation B, LSTM predicts a sequence of relations $y_1, y_2, ...y_t...$. The predicted relation y_t at time t is computed by:

$$s_t = BGRU(s_{t-1}, y_{t-1}, B); \tag{5}$$

$$p(y_t = l_j | y_1, .., y_{t-1}, B) = \frac{exp(l_j)Ts_t}{\sum_{l_i Ts_t}} \tag{6}$$

where y_t is the predicted relation at time t, l_i is the i-th relation, L is the relation set, T is a transformation matrix and s_t is the hidden state computed.

Since GRU predicts the relations of an entity pair as a series of single relations (in a sequential manner) by the probability of a relation conditioned on previously observed relations, which can model the conditional dependencies among these relations. Moreover, the predicted relation at each time step is also used as the input of the next time step, which can provide prior knowledge for the prediction of the next relation.

4.3 Attention Mechanism

After a relation is predicted at each time step, our model should pay more attention to those sentences whose corresponding relations have not been predicted. Therefore, we incorporate the attention mechanism into our model to dynamically adjust the bag representation, which reduces the impact of sentences whose corresponding relations have been predicted, and highlights the sentences which have not been covered. The bag representation at time t is computed by:

$$B_t = \sum_{i=1}^{n} \beta_{ti} x_i; \ \beta_{ti} = \frac{exp(\eta(s_{t-1}, x_i))}{\sum_{k=1}^{n} exp(\eta(s_{t-1}, c_k))}, \tag{7}$$

where β_{ti} is the weight of each sentence, η is a neural network. The score is based on the GRU hidden state s_{t-1} and the i-th sentence embedding x_i.

5. EXPERIMENTS

5.1 Dataset and Evauation Metrics

We use the same dataset(NYT10) as in [Lin et al.,2016]. NYT10 is originally released by the paper ”Sebastian Riedel, Limin Yao, and Andrew McCallum. Modeling relations and their mentions without labeled text.”Pre-Trained Word Vectors are learned from New York Times Annotated Corpus (LDC Data LDC2008T19), which should be obtained from LDC (https://catalog.ldc.upenn.edu/LDC2008T19).

5.2 Experimental Settings

Following previous work, we tune our models using three-fold validation on the training set. We use a grid search to determine the optimal parameters and select learning rate λ for SGD among $\{0.1,0.01,0.001,0.0001\}$, the sentence embedding size n $\in \{50, 60, ..., 300\}$, and the batch size B among $\{40,160,640,1280\}$.

For other parameters, since they have little effect on the results. For training, we set the iteration number over all the training data as 50. In the following Table we show all parameters used in the experiments.

Parameter Settings	
Sentence embedding size	250
Word dimension	50
Position dimension	5
Batch size	160
Learning rate	0.01
Dropout probability	0.3

5.3 Effect of Sentence Number

Since the superiority of our selective attention lies in the entity pairs containing multiple sentences, we compare the performance of CNN/PCNN+ONE, CNN/PCNN+AVE and CNN/PCNN+ATT on the entity pairs which have more than one sentence. And then we examine these three methods in three test settings:

- One: For each testing entity pair, we randomly select one sentence and use this sentence to predict relation.
- Two: For each testing entity pair, we randomly select two sentences and proceed relation extraction.
- All: We use all sentences of each entity pair for relation extraction.

Note that, we use all the sentences in training. We will report the P@100, P@200, P@300 and the mean of them for each model in held-out evaluation. From the following table, we can see that:

Test Settings	One			
P@N(%)	100	200	300	Mean
CNN+ONE	60.8	60.7	53.8	60.9
CNN+AVE	75.2	67.2	58.8	67.1
CNN+ATT	76.2	65.2	60.8	67.4
PCNN+ONE	73.3	64.8	56.8	65.0
PCC+AVE	71.3	63.7	57.8	64.3
PCNN+ATT	73.3	69.2	60.8	67.8
BGRU+ATT	81.0	75.5	70.0	75.5

Test Settings	Two			
P@N(%)	100	200	300	Mean
CNN+ONE	70.3	62.7	55.8	62.9
CNN+AVE	68.3	63.2	60.5	64.0
CNN+ATT	76.2	65.7	62.1	68.0
PCNN+ONE	70.3	67.2	63.1	66.9
PCC+AVE	73.3	65.2	62.1	66.9
PCNN+ATT	77.2	71.6	66.1	71.6
BGRU+ATT	82.0	79.0	73.6	78.2

Test Settings	All			
P@N(%)	100	200	300	Mean
CNN+ONE	67.3	64.7	58.1	63.4
CNN+AVE	64.4	60.2	60.1	60.4
CNN+ATT	76.2	68.6	59.8	68.2
PCNN+ONE	72.3	69.7	64.1	68.7
PCC+AVE	73.3	66.7	62.8	67.6
PCNN+ATT	76.2	73.1	67.4	72.2
BGRU+ATT	87.0	81.5	77.0	81.8

(1) Compared with both CNN and PCNN models, our BRRU+ATT+hierarchical relation outperforms the plain model. By taking advantage of the relation hierarchy, our models can learn better about long-tail relations via correlation information among relations. We also observe that even our hierarchical GRU model presents a better performance than the plain PCNN model. This shows the power of relation hierarchies, which makes attention plus relation hierarchies model outperforms the simple attention model on those long-tail relations.

(2) For both CNN and PCNN, the ATT method achieves the best performance in all test settings. It demonstrates the effectiveness of attention mechanism for multi-instance learning.

(3) For both CNN and PCNN, the AVE method is comparable to the ATT method in the One test setting. However, when the number of testing sentences per entity pair grows, the performance of the AVE methods has almost no improvement. It even drops gradually in P@100, P@200 as the sentence number increases. The reason is that, since we regard each sentence equally, the noise contained in the sentences that do not express any relation will have negative influence in the performance of relation extraction.

(4) CNN+AVE and CNN+ATT have improvements compared to CNN+ONE in the ONE test setting. Since each entity pair has only one sentence in this test setting, the only difference of these methods is from training. Hence, it shows that utilizing all sentences will bring in more information although it may also bring in some extra noises.

5.4 Effect of Hierarchical relation

To demonstrate the effects of the hierarchical relation, we empirically compare different methods through held-out evaluation. We

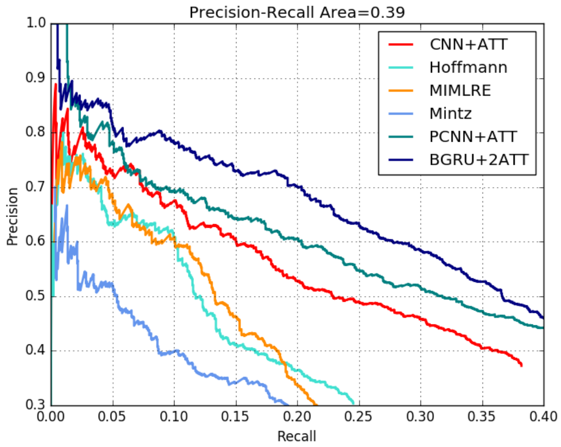

Fig. 4. Performance comparison of proposed model and traditional methods

select the CNN model proposed in (Zeng et al.,2014) and the PCNN model proposed in (Zeng et al., 2015) as our sentence encoders and implement them by ourselves which achieve comparable results as the authors reported.

Fig. 4 shows the precision/recall curves for each method. We can observe that: (1) BRRU+ATT+hierarchical relation significantly outperforms all feature-based methods over the entire range of recall. When the recall is greater than 0.1, the performance of feature-based method drop out quickly. In contrast, our model has a reasonable precision until the recall approximately reaches 0.3. It demonstrates that the human-designed feature cannot concisely express the semantic meaning of the sentences, and the inevitable error brought by NLP tools will hurt the performance of relation extraction. In contrast, CNN/PCNN+ATT which learns the representation of each sentences automatically can express each sentence well. (2) BRRU+ATT+hierarchical relation performs much better as compared with PCNN+ATT over the entire range of recall. It means that the hierarchical relation pinpoint more accuracy to the relation. Hence, the performance of our model can be further improved if we have a better sentence encoder.

6. CONCLUSIONS AND FUTURE WORKS

In this paper, we develop hierarchical relation extraction with GRU and sentence-level selective attention. Our model can make full use of all informative sentences and alleviate the wrong labelling problem using hierarchical relation mechanism for distant supervised relation extraction. In experiments, we evaluate our model on relation extraction task. The experimental results show that our model significantly and consistently outperforms state-of-the-art feature-based methods and neural network methods.

In the future, we will explore the following directions: Our model incorporates multi-instance learning with neural network via hierarchical relation and instance-level selective attention. It can be used in not only distant supervised relation extraction but also other multi-instance learning tasks. We will explore our model in other area such as text categorization. GRU is one of the effective neural networks for neural relation extraction. Researchers also propose many other neural network models for relation extraction. In the future, we will incorporate our technique with those models for relation extraction.

7. REFERENCE

Guoliang Ji, Kang Liu, Shizhu He, Jun Zhao, et al. 2017. Distant supervision for relation extraction with sentence-level attention and entity descriptions. In AAAI, pages 30603066.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In Proceedings of ACL, pages 21242133.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In Proceedings of NAACL, pages 7484.

Cicero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying relations by ranking with convolutional neural networks. In Proceedings of ACL-IJCNLP, pages 626634.

Yan Xu, Lili Mou, Ge Li, Yunchuan Chen, Hao Peng, and Zhi Jin. 2015. Classifying relations via long short term memory networks along shortest dependency paths. In Proceedings of EMNLP, pages 17851794.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In Proceedings of EMNLP, pages 17531762.

Dongxu Zhang and Dong Wang. 2015. Relation classification via recurrent neural network. arXiv preprint arXiv:1508.01006.