Extracting Personae Interactive Relation in Chinese Microblog Based on an Improved Dependency Trigram Kernel

Anzhen Yang¹,a, Yajun Du¹,b, Qingrui Meng²

¹School of Computer and Software Engineering, Xihua University, Chengdu Sichuan 610039, China
²Tibet FeiYue Intelligent Technology CO., LTD, China
ayanganzhen1226@126.com, duyajun@mail.xhu.edu.cn

Keywords: persons relation extraction; dependency trigram kernel; microblog

Abstract: As the rapid development of social media, microblog has arrested a lot of attention. However, the automatic information extraction task of the microblog is relatively rare because the microblog text is quite complex and irregular. In this paper, we utilized dependency trigram kernel to construct persons relations extraction model. Firstly, we described the dependency trigram kernel (DTK) for relations extraction. Secondly, we used words semantic similarity tool HowNet to improve semantic similarity of dependency trigram's words. And then we proposed "(POS,GR) (Part of Speech,Grammatical role)" pair to improve dependency trigram's words syntax similarity. Finally, we evaluated the validity of the relation extraction model by experimenting, and the results of experiment show that the F-value of our improved DTK is higher than original DTK for microblog persons relation extraction.

1 Introduction

With the popularization of microblog, a large number of microblog texts were generated and updated within a day. Naturally, users can't get satisfactory results for searching information in microblog. Microblog persons entity relations can construct microblog knowledge graph to extend microblog persons knowledge search. It is also used for building microblog question-answer system. In the task of relation extraction, ACE08 (Automatic Content Extraction 2008) [1] defined seven relation types between entities. After ACE proposed these relation definition, many researchers have been proposed various methods for relation extraction. These methods could classify into three categories: rule based model, features based model and kernel based model. Brin et al [2] proposed a rule based relation extraction system named DIPRE (Dual Iterative Pattern Relation Expansion). It was a bootstrap method, and they manual tagged a large number of words-syntax patterns for extracting relations. Agichtein et al [3] improved DIPRE system and proposed another system named Snowball. They tagged the type of named entities. The precision of the system was improved by limiting the entities types, and then they appended confidence degree estimate model to system. It took of relations with lower confidence degree from relation seed sets. The weakness of the rule based model is that the corpus requires constant manual tag. To solve this problem, the feature based models were proposed. This kind of model reduces the work of the manual tag by transforming text information into linear feature vectors. Zhang et al [4] combined syntactic context information, entity types, vocabulary and part of speech features with maximum entropy model for extracting relation in the filed of tourism. Claudio et al [5] considered the entity features such as context and verb etc. And they utilized SVM to classify relations. Tymoshenko [6] based on Claudio's method, used the Cyc knowledge base to extend entity features. However, a sentence is usually expresses as a tree structure, so the linear feature based method can't work effectively. For this reason, the kernel based model was proposed to deal with tree structure of sentences. Firstly, Kernel based model constructed high dimensions vectors for relation features, and then it used a kernel function to calculate the similarity among these vectors. Finally, the model used classifier to categorize these vectors. Zelenko et al [7], proposed a kernel function for phrase trees, and they designed a dynamic programming algorithm to extract relations. Their experiment achieved satisfactory results on news corpus. Culotta and Sorensen [8] continued Zelenko's study. They estimated the similarity among relations by utilizing incremental dependency trees. Their experiment extracted
5 kinds of relations on ACE2003 corpus, and the F-value achieved 45.8%. Zhou et al [9] proposed a convolution tree kernel. The convolution tree was consisted of rich syntax rules and semantic relations. Their experiment on the Automatic Content Extraction/Relation Detection and Characterization (ACE RDC) corpora shows that their method outperforms other state-of-the-art methods.

The rest of this paper is organized as follows. In section 2, we introduced the idea of dependency trigram kernel. In section 3, we improved the dependency trigram set similarity function. Firstly, we utilized HowNet to calculate words semantics similarity, and then we proposed (POS,GR) pairs to measure syntax similarity. In section 4, we described the detailed process of persons relation extraction. The experiment is built and analyzed the effect of this model. Finally we summarize this article and prospect to the future works.

2 Relate work

The dependency trigram is a learning algorithm based on support vector machine (SVM). Choi and Kim [10] propose DTK to extract people social relations from ACE corpus and Korean news. They divide relation extraction process into two phases, firstly, is select sentences which are contain relations, second is identify the relation name. Both phases are utilize DTK to train data. The DTK can transform a sentence into some dependency trigrams. Given a sentence S, and the dependency tree of S shown as follows:

![Figure 1. The example of a dependency tree.](image)

According to the tree, they design two kernel functions for selecting sentences containing social relations model and extracting names of social relations model respectively. Firstly, the kernel function for selecting sentence shown as follows:

\[
K(A_T, B_T) = \sum_{i=1}^{n} \max\{S(A^i_T, B^i_T), S(A^i_T, B^2_T)\ldots S(A^i_T, B^n_T)\}
\] (1)

Where A, B represent two sentences, and A^i_T, B^j_T represent two dependency trigrams in sentence A, B. S(A^i_T, B^j_T) is similarity measures function between two dependency trigram. The dependency trigram is core component for calculate similarity, it contains various features of a sentence such as words literal meaning, syntax and part of speech (POS). Kernel function (1) considers all dependency trigrams in two sentences to find out whether a new sentence contains relation. After selecting sentences, the relation name kernel function is proposed, shown as follows:

\[
K(A^i_T, B^j_T) = S(A^i_T, B^j_T)
\] (2)

This function only considers each dependency trigram whether contains a relation name key word. Finally, they use support vector machine (SVM) to learn manual tagged corpus, and then extract relations from news text.

3 Extract persons interaction relationship in microblog

In order to make DTK suitable for dealing with microblog text, we improved the similarity function among dependency trigram sets. Firstly, we utilize HowNet to calculate words semantics similarity in dependency trigrams. Secondly, we proposed (POS,GR) pairs to represented words syntax similarity. Finally, we proposed a new function to measure words semantics, syntax weight factors.
3.1 improved dependency trigram similarity for word semantics. The dependency trigram set similarity function is a core part of the dependency trigram kernel function. The function $S(A_T^i, B_T^j)$ considered three words features in a sentence, they expressed these features as binary value. The binary value of words semantic feature usually returns 0 in microblog text. Hence we improved the words semantic feature by using digit value which is a value between 0 and 1. The digit value guaranteed the enough contribution of words semantic feature. We used a natural language process tool which is named HowNet to obtain the words semantic similarity. HowNet built a dictionary of words, and calculate the distance between words to gain the words semantic similarity. In a given dependency trigram $w_c \rightarrow w_k \leftarrow w_l$, $w_l, w_r$ called left word and right word respectively, $w_c$ called center word. We proposed following function to improve the words semantic feature in DTK:

$$
\text{simWords}(A_T^i, Tag_T^j) = \alpha_i \times \frac{\text{sim}(A_T^i, Tag_T^j) + \text{sim}(A_T^i, Tag_T^j)}{2} + \beta_i \times \text{sim}(A_T^i, Tag_T^j)
$$

(3)

where the $A_T^i$ represent a new dependency trigram for sentence $A$, and $Tag_T^j$ represent all the dependency trigrams for manual tagged sentences. $w_l, w_r, w_c$ are left, right and center word in a dependency trigram. Function $\text{sim}(x, y)$ used the HowNet to measure words semantic similarity. The parameters $\alpha_i, \beta_i$ represent side words weight and center word weight respectively. If side words are microblog person entity and center words are verb, the weight will be higher than other words. So we designed the function of weight factor $\alpha_i, \beta_i$, it shown as follows:

$$
\begin{align*}
\alpha_i &= \frac{\text{NumEntity}(A_T^i, A_T^i) + 1}{\text{NumEntity}(A_T^i, A_T^i) + \text{NumVerb}(A_T^i) + 2} \\
\beta_i &= 1 - \alpha_i
\end{align*}
$$

(4)

In Eq (4), the function $\text{NumEntity()}$ returned the number of entities in side words, and $\text{NumVerb()}$ returned the number of verbs in center words.

3.2 improved dependency trigram similarity for word syntax. Next, we calculated two other features of dependency trigram set namely part of speech (POS) and grammatical role (GR). POS is a word belongs to adjective, verb or noun and so on. GR means a word is belongs to object, subject or predicate. The words of POS and GR are constant in a sentence, and we considered the POS and GR as a whole. So that the POS and GR features could be represented as (POS,GR) pairs.

$$
\text{SimSyntax}(A_T^i, Tag_T^j) = \alpha_i \times \frac{\text{syn}(A_T^i, Tag_T^j) + \text{syn}(A_T^i, Tag_T^j)}{2} + \beta_i \times \text{syn}(A_T^i, Tag_T^j)
$$

(5)

Where $A_T^i$ represent a new dependency trigram from a microblog sentence. $Tag_T^j$ represent all the dependency trigrams from manual tagged microblog sentences. $A_T^i$, $Tag_T^j$ represent words (POS,GR) pairs of a new dependency trigram. Function $\text{syn}(X, Y)$ means that the probability of $X$ in $Y$. The parameters $\alpha_i, \beta_i$ are weight factors which used formula (4) to measure their value.

3.3 Integrate semantics and syntax features for dependency trigram. After get semantics and syntax similarity of a dependency trigram, we design two weight factors $\gamma, \delta$ to weight semantics and syntax features, the formula shown as follows:

$$
S(A_T^i, Tag_T^j) = \gamma \times \text{SimWord}(A_T^i, Tag_T^j) + \delta \times \text{SimSyntax}(A_T^i, Tag_T^j)
$$

(6)

In Eq (6), $\gamma, \delta$ represent words literally and syntactic features weight factors respectively. We used the information entropy to calculate words literally weight $\gamma$, the equation shown as follows:

$$
E = - \sum_{x \in \text{words}} p(x) \log_2 p(x)
$$

(7)
where the \( p(x) \) is the probability of word \( x \) in all corpus. Due to words syntactic features like POS and GR always have constant relation, so we used the mutual information entropy (MIE) to represent the syntactic features weight \( \delta \), and the equation shown as follows:

\[
MIE = \sum_{x \in \text{pos}} \sum_{y \in \text{gr}} p(x, y) \log_2 \left( \frac{p(x, y)}{p(x)p(y)} \right)
\]  

(8)

In Eq (8), \( p(x, y) \) is the probability of a word POS-GR pairs in whole corpus, \( p(x) \) is the probability of a word POS, and \( p(y) \) is the probability of a word GR. After defined \( E \) and \( MIE \), we could get the equality of \( \gamma \) and \( \delta \), shown as follows:

\[
\begin{align*}
\gamma &= \frac{E}{E + MIE} \\
\delta &= 1 - \gamma
\end{align*}
\]  

(9)

In Eq (9), \( E \) is the entropy of words literally and \( MIE \) the mutual information entropy of POS and GR.

4 Experiments and results

In order to evaluate the proposed system experimentally, we crawled lots of real data about personage topics in tencent microblog. We first need to extract the persons entities in these texts, and divided these sets into 2 parts. The first part is manual tagged sets which need manual tagging the relations name between persons entities, and the sets included 12678 posts. The Second part is test sets which need system recognizing the relations between person entities, and the test sets included 3678 posts. After collecting the corpus, we conducted the experiment of relations extractions, and we compared the original DTK and improved DTK. We crawled 5 different Chinese microblog topics, firstly, we used LTP (Language Technology Platform Cloud) to pre-process these microblog text, we can get the linguistic information including dependency tree, person named entities, POS and grammatical role. Secondly, we extracted dependency trigram for sentences selecting from manual tagged corpus and test corpus, and then we used improved DTK to calculate the similarity of dependency trigram, in this step, we thought a sentence should have a persons relation when the similarity was greater than 0.7. Finally, we extracted for selecting relation names, and used improved DTK to confirm the relation names when a relation name dependency trigram similarity was greater than 0.65, and we chosen three evaluation criterions \( P \) (precision), \( R \) (recall), and \( F \) (F-value) for evaluating experimental result. We totally extracted 875 relations from 3678 microblog posts, and the results of extraction shown in Fig 2, it is easy to see that our improved DTK’s F-value is approximately 10% higher than original DTK.

![Figure 2. The F-value of original DTK and improved DTK.](image-url)
Conclusions and future work

In this paper, we improved DTK by proposing the POS-GR pairs to measure words syntax information and using HowNet to measure words semantics in trigrams. The processes of extraction divide into two step, first step is select sentences which contain a relation, and second step is confirm relation name in these sentences. The results show that our improved DTK is more effective than original DTK for Chinese microblog persons relations extraction.

Acknowledgment

This work is supported by the National Nature Science Foundation of China (No. 61271413 and 61472329), Tibet Museum of Natural Science Digital key technology Research and Integration, National Key Technology R&D program (No. 2011BAH26B01), The Graduate Student Innovation Foundation of Xihua University under Grant (No. ycjj2015186), the Science and Technology Innovation Seedling Project of Sichuan (Grant No.2015080, No.2015074, No.2015075), all support is gratefully acknowledged.

References


