WordNet Based Multi-Way Concept Hierarchy Construction from Text Corpus

Ding Tu, Ling Chen & Gencai Chen
College of Computer Science, Zhejiang University
Hangzhou 310027, P.R. China
{10921014, lingchen & chengc}@zju.edu.cn

Abstract

In this paper, we propose an approach to build a multi-way concept hierarchy from a text corpus, which is based on WordNet and multi-way hierarchical clustering. In addition, a new evaluation metric is presented, and our approach is compared with 4 kinds of existing methods on the Amazon Customer Review data set.

Introduction

With a concept hierarchy, people can organize information into categories and concentrate on a particular aspect of the information. There have been many works proposed in literature to construct a concept hierarchy from corpus. Some works focus on using statistical methods to build a concept hierarchy (e.g. Andreas et al. 2005). Another kind of methods uses predefined rules to get particular kinds of relations (e.g. Wu et al. 2012). Besides that, some works use external semantic dictionaries.

These works mainly concern with the extraction of the semantic relations between the words, but the quality of the words is not mentioned. And traditional clustering methods generally output hierarchies in a binary tree form, which is not the natural way to represent knowledge. In this paper, we propose a WordNet based method to build a multi-way concept hierarchy from a text corpus, and our contributions are: 1) apply a multi-way hierarchical agglomerative clustering algorithm to organize the candidate concept words extracted in the corpus; 2) propose a new metric to evaluate the similarity of the organization of two hierarchies with same leaf nodes.

Framework

The overall process can be divided into four steps.

Text Preprocessing

The text preprocessing step is to get high quality text document. It can be divided into three phrases. The first phrase aims to extract the text content of a document set in different formats. The second phrase is to eliminate unimportant words in the extracted text content and index them. All the unimportant words in the text content, e.g. prepositions and conjunctions, would be ignored and not be indexed. In the third phrase, we use the index to transform every document into a document vector that can be used in the next steps.

Topic Model based Keywords Extraction

We argue that all the words in the concept hierarchy of a text corpus should support people to understand the content of the documents and represent a particular aspect of the corpus. Therefore, we use latent Dirichlet allocation (LDA) (Blei et al. 2003) to model a text corpus. LDA is an unsupervised machine learning method based on the “bag of words” assumption, and it has got a great success in finding latent topics in a text corpus. Our goal is to extract the most typical words in every latent topic, with which we can find the aspects that those latent topics care about. One result of the LDA is the topic word distribution $P(w|z)$, which represents the distribution of word w for topic z. We choose the top n words that have the largest probability in a topic as the candidate concept words of the topic.

Semantic Relation Extraction

To find out which aspect of the content those candidate concept words belong to, the program should know the semantic meanings of the candidate concept words. Our method extracts the semantic relations with the help of WordNet. The reason is that in a specific corpus (e.g. customer reviews or microblogs), some relations cannot be extracted from the content using statistical methods (Xing et al. 2012). Using WordNet can solve this problem in most cases, and we do not have to build a predefined knowledge base for every specific corpus.

After using an automated word sense disambiguation tool, we can get the most possible sense for each candidate word. With the hypernym relations in WordNet, we might find the common ancestor of two words. For two concept words w_1, w_2 with common ancestor w_3 in the WordNet, the *semantic distance with depth* of the two words is $SDD(w_1, w_2)=(L(w_1, w_3)+L(w_2, w_3))/n^d$. $L(w_1, w_3)$ is the length of the path between w_1 and w_3. n is a real number larger than 1, and d is the depth of w_3 in WordNet. This is based on the assumption that for two word pairs with a same distance, the pair locates deeper in the semantic tree are semantic closer. For words that do not have a common ancestor, we set the distance to a large value.

Hierarchical Clustering

Traditional hierarchical clustering algorithms may not show the relations between words appropriately, for organizing words into a binary tree will change the relative closeness of word pairs. Therefore, we apply a multi-way hierarchical agglomerative clustering algorithm (Kuo et al. 2005) to cluster candidate concept words.

The algorithm has three kinds of merge to support the multi-way clustering. First, it puts all input concept words as singleton concept clusters. Then, the algorithm turns into the iteration phrase. For every iteration, it selects the concept pair (c_A, c_B) with the smallest distance in all pairs. If the inter-concept distance between c_A c_B is larger than a threshold τ, it makes them as the sub-concepts of a new concept. The threshold τ is related to the standard deviation of the input concept set. If the distance between two concepts is smaller than τ, then it compares their density. We use $ICD_{\text{avg}}(c_A)$ to represent the average inner-concept distance of c_A. We assume that the density of c_A is larger than c_B, i.e. $ICD_{\text{avg}}(c_A)>ICD_{\text{avg}}(c_B)$. If $ICD_{\text{avg}}(c_A)/ICD_{\text{avg}}(c_B)$ is smaller than σ, it puts all sub-concepts of c_A and c_B under a new concept. Else, c_B becomes a sub-concept of c_A. σ is a real number larger than 1. If $ICD_{\text{avg}}(c_A)/ICD_{\text{avg}}(c_B)$ is smaller than σ, the densities of two concepts are similar. This process iterates until all concepts are merged into one.

Performance Evaluation

We evaluate the construction process of our method through comparing the similarity between the result hierarchy and a target hierarchy. The target hierarchy is manually constructed, using the same candidate concept set with the result hierarchy.

Evaluation Metric

Our similarity model is modified from the evaluation method using core ontology (Andreas et al. 2005), which does not consider the labeling issue. Our basic idea is that the structure of a tree can be split to paths form root to leaf nodes. The distance of two concept hierarchies can be measured by computing the similarity of paths from root to the same leaf nodes.

We define the similarity of two concepts c_1 and c_2 in concept hierarchies H_1 and H_2 as $Sim_{\text{concept}}(c_1, H_1, c_2, H_2) = |UC^*(c_1, H_1) \cap UC^*(c_2, H_2)|/|UC^*(c_1, H_1) \cup UC^*(c_2, H_2)|$. Here, $UC^*(c_1, H_1)$ represents all leaf concept nodes of c_1 in H_1. For two paths consists of two sets of concept nodes U and W, we need to find a matching $M_{\max}(p_1, p_2)$ from the subset of $U \times W$, which should satisfy the following constraints: (1) all pairs in M should be in order, that is for two pairs (u_1, w_1) and (u_2, w_2) , if $u_1<u_2$, then $w_1<w_2$, $<$ means one node is in front of another node on the path; (2) all the nodes appear in the pairs at most once; (3) the nodes of the smaller set should all be paired; (4) the sum similarity of the pairs should be the maximum value. With the matching M, the similarity of paths is:

$$Sim_{\text{path}}(p_1, p_2)=a\sum_{(u,\, w)\in M}Sim_{\text{concept}}(u, H_1, w, H_2) + b(2min(L(p_1),\ \mathrm{L}(p_2))/(L(p_1)+L(p_2)))$$

The similarity of two hierarchies can be computed as the average value of all the similarities between the paths of the same leaf concept nodes.

Experiments and Results

We evaluate our approach in five domains with the product review data crawled from Amazon (Jindal et al. 2010). We set σ as 1.1-1.9, a as 0.8, b as 0.2. Five different combinations are compared in our experiment: two use WordNet but with different clustering methods: multi-way clustering method (MWN), hierarchical agglomerative (AWN); three use co-occurrence relations to extract semantics (bisection kmeans (BIS), hierarchical agglomerative (AGG), and multi-way clustering (MCO)).

Table 1: the result of four combinations in five domains

	BIS	AGG	MCO	AWN	MWN
Notebook	0.581	0.611	0.738	0.674	0.730
Pen	0.558	0.569	0.637	0.625	0.685
Stroller	0.621	0.674	0.807	0.781	0.802
TV	0.595	0.62	0.715	0.631	0.698
Watch	0.601	0.605	0.665	0.807	0.849

The results are presented in Table 1, and it shows that the combination using multi-way clustering and WordNet get the highest similarity score.

Ackonwledgements

This work was funded by the Ministry of Industry and Information Technology of China (No.2010ZX01042-002-003-001), Natural Science Foundation of China (Nos. 60703040, 61202282), and Science and Technology Department of Zhejiang Province (Nos. 2007C13019, 2011C13042).

References

Cimiano P.; Hotho A.; Staab S. 2005. Learning concept hierarchies from text corpora using formal concept analysis. *Journal of Artificial Intelligence Research* 24(1): 305-339.

Blei, D. M.; Ng, A. Y.; and Jordan M. I. 2003. Latent Dirichlet Allocation. *The Journal of machine Learning research* 3: 993-1022.

Jindal, N.; Liu B.; and Lim E. P. 2010. Finding Unusual Review Patterns Using Unexpected Rules. *In Proc. of CIKM-10,* 1549-1552.

Kuo, H.; and Huang J. 2005. Building a Concept Hierarchy from a Distance Matrix. *Intelligent Information Processing and Web Mining*. 31: 87-95.

Wu, W.; Li H.; Wang H.; and Zhu K. Q. 2012. Probase: A Probabilistic Taxonomy for Text Understanding. *In Proc. of SIGMOD-12*, 481-492.

Chen X.; Li L.; Xiao H.; Xu G.; Yang Z.; and Kitsuregawa M. 2012. Recommending Related Microblogs: A Comparison Between Topic and WordNet Based Approaches. *In Proc. of AAAI-12.*