A New Semantic Similarity Measurement Based on HowNet Concept Tree

Xiaohua Guo[1,a], Xinhua Zhu[1,b], Fei Li[1,c], Qi Li[1,d]

[1]Guangxi Key Lab for Multi-source Information Mining & Security, Guangxi Normal University, Guilin 541004

[a] 15577430403@163.com,[b]zxh429@263.net, [c]312078417@qq.com, [d]804995755@qq.com

Keywords: Concept Tree, HowNet, Semantic Similarity, Path, Depth.

Abstract. This paper puts forward a new depth & path-based semantic similarity method to improve the existing meaning-based approaches in HowNet. Firstly, a complete concept tree is constructed on the sememe tree according to the concept definitions in HowNet. Then an improved depth & path algorithm was put forward, in which five adjustable parameters are used to compute the depth and path of concept in concept tree. This method avoids the computation process of complicated meaning similarity and is more intuitive and efficient. The experiment shows that proposed algorithm has achieved an excellent level comparing with the existing word similarity algorithms.

Introduction

Word semantic similarity has been widely used in machine translation, information retrieval, text mining, word sense disambiguation and intelligence tutoring. Currently, WordNet is internationally main world knowledge base to measure semantic similarity, and the word similarity research on WordNet has reached a higher level internationally. In China, most scholars mainly use HowNet [1] or CiLin to measure semantic word similarity. HowNet is devoted to machine translation between Chinese and English, in which concept explanation includes the bilingualism of Chinese and English. So HowNet and WordNet can promote each other on word similarity research. However, most of HowNet-based word semantic similarity measurements in China are implemented by computing meaning similarity between two concepts. But the meaning similarity algorithm requires measuring the similarity between sememe groups in the concept definition expression, which is complicated, unintuitive and inefficient. To avoid the complicated meaning computation, this paper puts forward a direct depth & path-based semantic similarity method by constructing a concept tree that contains all concepts in HowNet. This method is simple, intuitive and high efficiency.

Breifing to HowNet

To begin with, HowNet is established by Mr Dong to realize machine translation between Chinese and English, which includes 62364 concepts in its 2000 version. After constant updating and developing, the structure of HowNet has been constantly improved. There are two important elements in HowNet: sememe and concept. Sememe is the smallest meaning unit, which is used to describe and explain other concepts. Concept is an explanation of the word, and a word may have multiple concepts. Concept is associated with sememe mainly by the semantic expression. Semantic expression is the main body of concept, which is used to explain concepts. Multiple sememes constitute the semantic expression combined with knowledge description symbol, so as to establish the connection with concepts.

Semantic expression is also called as the definition of concept and can be abbreviated to DEF, and a concept of each word is usually made up of a quad in HowNet: < W_X = concept name, E_X = concept example, G_X = concept speech, DEF = concept definition>.

There are more than 1500 sememes in HowNet 2000 version. All the basic sememes form a hierarchy tree (as shown in Fig. 1).

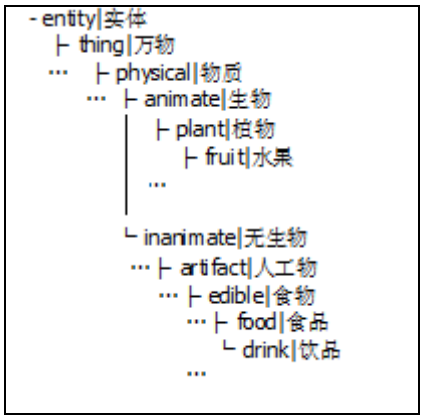

Fig. 1: Hierarchy tree of sememe

Relevant Semantic Similarity Approaches

Meaning-based approaches in HowNet. In the study of word similarity based on HowNet, Qun Liu et al. made contributions first. They measure word semantic similarity by computing meaning similarity between two concepts in HowNet, which requires measuring the similarity between sememe groups in the DEF of concept. It's expressed by Eq. 1:

$$Sim(c_1, c_2) = \sum_{i=1}^{4} b_i sim_i(c_1, c_2) \tag{1}$$

For the calculation convenience, Qun Liu divided the DEF into four parts:

The first basic sememe: the first sememe in the concepts' DEF, which is used to describe the main semantic features.

Other basic sememe: the sememes behind the first sememe and without any logic description symbol in the DEF of concept.

Relational sememe: the sememes with "=" in the DEF, which is used to describe the relationship between defined concept and the other one.

Relational symbol sememe: the sememes with logic description symbol. These logical symbols are shown as follows: ,~^+&@#%$*?![]{}(). Each symbol expresses a special relationship. For example, "," means "and" between multiple attributes, "#" means "relevant" between sememes, etc.

In Eq. 1, $sim_1(c_1,c_2)$, $sim_2(c_1,c_2)$, $sim_3(c_1,c_2)$, $sim_4(c_1,c_2)$ are the similarity between the first basic sememes, other basic sememes, relational sememes and relational symbol sememes for two concepts' DEF respectively, and β_i ($1 \le i \le 4$) is adjustable parameters, and $\beta_1+\beta_2+\beta_3+\beta_4=1$ ($\beta_1 \ge \beta_2 \ge \beta_3 \ge \beta_4$). Among them, the similarities between two sememes are measured by Eq. 2:

$$Sim(p_1, p_2) = \frac{s}{d+s} \tag{2}$$

Where p_1 and p_2 are two sememes, and d is the shortest path length between p_1 and p_2 in the sememe tree, which is a positive integer, and σ is an adjustable parameter, of which value is 1.6 in the paper of Qun Liu .

Path & Depth-based Approaches in WordNet. Wu and Palmer first put forward word similarity computation method based on path and depth in the field of WordNet. Thereafter, Liu et al.[2] also put forward a word similarity computation method based on HowNet, which is an optimized method based on method of Wu and Palmer. Its computational formula is shown as Eq. 3:

$$sim_{Liu}(c_1, c_2) = \frac{a \times d}{a \times d + b \times p} \tag{3}$$

Where d is the depth of the lowest common hypernym for words c_1 and c_2 in the taxonomy hierarchy, p is the shortest path length between words c_1 and c_2 in the taxonomy hierarchy, α and β are the adjustable parameters for depth and path respectively ($0<\alpha, \beta<1$).

Liu et al. [2] think that, when α=0.5, β=0.55, the Eq. 3 is the best.

Constructing Concept Tree

Real Node and Virtual Node in Concept Tree. In fact, the knowledge description symbol in HowNet is the simplified description logic (DL)[3] . For example, the DEF of concept "amateur" is "amateur == human, *FondOf, #WhileAway", which is equal to the concept definition in description logic:

$$\text{Amateur} == \text{human} \cap \exists \text{relevant concept.fondof} \cap \exists \text{agent.whileaway}$$

In the DEF of HowNet concept, the description symbol of "," can be transferred into the operator "∩" in description logic, and "*FondOf", "#WhileAway" can be respectively transferred into the relational constraints [4] with existential quantifier "∃" in the description logic. In the concept definition of "amateur", "human" is the first basic sememe, "relevant concept" and "agent" are two attributes of "human", and "fondof" and "whileaway" are two attribute values. The explanation of the concept definition for the above "amateur" is: "amateur" is a human who is fond of something and can be used for the agent of "whileaway".

In constructed concept tree, we collectively call all exsiting concepts and sememes in HowNet as the real nodes in concept tree, and the intersection or union between the first basic sememe and relational constraints in the concept definition as the virtual node in concept tree. In the above example of "amateur", "amateur" and "human" are real nodes, and "human ∩ ∃relevant concept.fondof" and "human ∩ ∃relevant concept.fondof ∩ ∃agent.whileaway" are virtual nodes.

Contrusting Concept Tree. In order to make all the basic sememes form a tree according to is-a relation, we add a virtual node as the root node to connect all category trees in HowNet, so as to constitute a complete sememe tree.

First, we determine the positions at the sememe tree for all first basic sememe of each DEF of concept. Then we transfer each other sememe in the DEF of concepts into relational constraints with "∃" in the description logic. We take them as the virtual nodes and place them under the sememe tree one by one. After all sememes in the DEF have been picked up, we place the defined concepts at the bottom of its first basic sememe or its virtual nodes that has been place at the tree and take it as a leaf node in the tree. Finally a concept tree is set up successfully, as shown in Fig. 2.

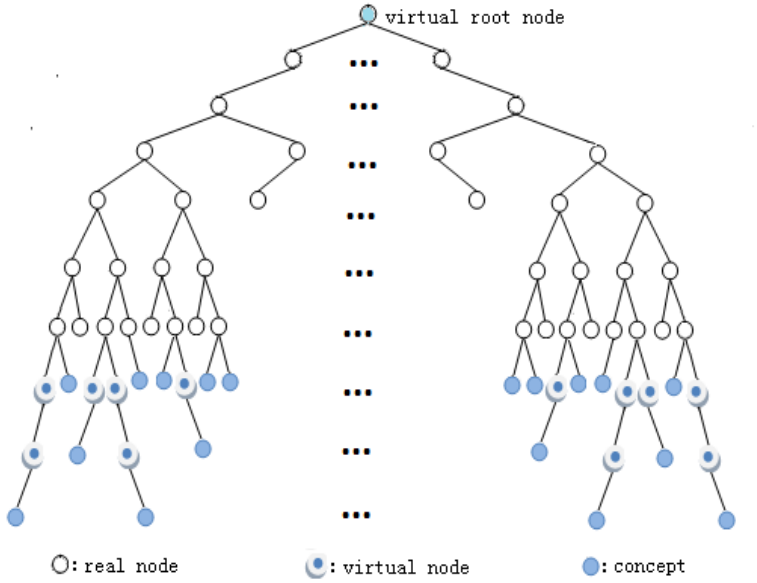

Fig. 2: Concept tree

Proposed Semantic Similarity Approach

Similarity Formula based on the HowNet Concept Tree. First, this paper adopts the method of Liu [2]. Formula is shown as Eq. 4:

$$sim(c_1,c_2)=\frac{a\times Deep\left(LCP(c_1,c_2)\right)}{a\times Deep\left(LCP(c_1,c_2)\right)+b\times Path(c_1,c_2)+InitialComp(c_1,c_2)} \tag{4}$$

Where $Deep(LCP(c_1,c_2))$ is the depth of the lowest common hypernym for concepts c_1 and c_2 in concept tree, and $Path(c_1,c_2)$ is the shortest path length between concepts c_1 and c_2 in concept tree, and α and β are the adjustable parameters for $Deep(LCP(c_1,c_2))$ and $Path(c_1,c_2)$ respectively. To avoid this situation that similarity value of non-synonymous concepts is 1, we introduce initial function $InitialComp(c_1,c_2)$ to improve it in the computational process.

Considering that we add a virtual root node in constructed concept tree, the similarity computed in Eq. 4 is multiplied by a cosine value for smoothing the similarity. In this way, we can get the final similarity computation formula between two concepts as Eq. 5:

$$FinalSim(c_1,c_2)=sim(c_1,c_2)\times Cos\left(\left(1-sim(c_1,c_2)\right)\times\frac{p}{2}\right) \tag{5}$$

Where $\mathrm{sim}(c_1,c_2)$ is the similarity result between concept c_1 and c_2 computed by Eq. 4, and $\mathrm{FinalSim}(c_1,c_2)$ is the final similarity between two concepts. When the lowest common hypernym for two concepts is root node, we give the final similarity between two concepts a very low value of 0.01.

Assuming that word w_1 has n_1 concepts and word w_2 has n_2 concepts, we put forward similarity computation formula between the words w_1 and w_2 as Eq. 6:

$$sim(w_1,w_2)=\begin{cases}\max\limits_{i=1\mathbf{L}n_1,\,j=1\mathbf{L}n_2}\{\mathrm{FinalS}im(c_{1i},c_{2j})\}, & w_1\neq w_2\\ 1\qquad\qquad\qquad\qquad , & w_1=w_2\end{cases} \tag{6}$$

Where $FinalSim(c_{1i},c_{2j})$ is the similarity value between the ith concept of word w_1 and the jth concept of word w_2.

In this paper, all path lengths and depths are not restricted to integer.

Computation of Path Length. We think real nodes are the major contributors to path length between two concepts, and virtual nodes are auxiliary contributors in concept tree. Therefore we put forward computational formula of path length as Eq. 7:

$$Path(c_1,c_2)=path\left(FS(c_1),\ FS(c_2)\right)+VirtComp(c_1,c_2) \tag{7}$$

Where $FS(c)$ is the first basic sememe in the DEF of concepts c, and $path(FS(c_1),\ FS(c_2))$ is the shortest path length between $FS(c_1)$ and $FS(c_2)$ in concept tree, and $VirtComp(c_1,c_2)$ is compensation of path based on virtual node.

Feng Li et al. [5] think that the basic sememe without a symbol in the DEF is a direct description for a concept, and relational sememe and relational symbol sememe with a symbol are the indirect description for a concept. To distinguish their influence on word similarity measurement, we introduce $VirtComp(c_1,c_2)$, of which computational formula as Eq. 8:

$$VirtComp(c_1,c_2)=Sum(SS)\times h+Sum(OS)\times l \tag{8}$$

Where $Sum(SS)$ is the sum of different relational symbol sememes' number and relational sememes' number in the DEF for two concepts, and $Sum(OS)$ is the sum of different other basic sememes' number in the DEF for two concepts, and η is the adjustable parameter for $Sum(SS)$, and λ is the adjustable parameter for $Sum(OS)$.

Computation of Depth. Considering different effects of sememes, all kinds of the same sememes also are multiplied by corresponding parameters. Its computational formula is shown as Eq. 9:

$$Deep(LCP(c_1,c_2)) = Deep(FS_1) + SameSum(SS) \times h + SameSum(OS) \times l + SearchLev(FS_1) \quad (9)$$

Where FS_1 is the first basic sememe in the DEF of concept c_1, and $Deep(FS_1)$ is the depth of FS_1 in concept tree, and $SameSum(SS)$ is the sum of the the same number of relational symbol sememes and relational sememes in the DEF for two concepts, and $SameSum(OS)$ is the sum of the same number of other basic sememes in the DEF for two concepts, and $SearchLev(FS_1)$ is the shortest path length between FS_1 and the lowest common hypernym for concept c_1 and concept c_2 in concept tree, and the value of η and λ are same with Eq. 8.

Computation of Initial Function. The computational formula of initial function is shown as Eq. 10:

$$InitialComp(c_1,c_2) = \frac{d}{SameVirt(c_1,c_2)+1} \quad (10)$$

Where δ is a adjustable parameter for initial function, and $SameVirt(c_1,c_2)$ is the same number of virtual nodes for concept c_1 and concept c_2 in concept tree.

Determination of Parameter. When measuring similarity between two sememes, or sememe and concept of which DEF has only the first basic sememe in the DEF of concept, we give path's compensation based on virtual node a value of 0, otherwise we will do nothing.

Given that the direct description is more important than indirect description, we think that η shouldn't be greater than λ. In addition, initial parameter δ only play the role of fine adjustment for experimental data, so δ shouldn't be too large. In the experimental process, after repeated comparison and parameter adjustment, we think α=0.40, β=0.88, δ= 0.01, η= 0.84, λ= 0.9 is reasonable.

Experiment and Analysis

Experimental Comparison. First of all, we adopt MC30 test set in relevant comparison experiment. Firstly, we build a connection between English and Chinese concepts documents according to HowNet 2000 version[1]. Then, using methods of Qun Liu, Feng Li and our method to measure similarity of MC30 data set based on HowNet 2000 version respectively. Finally, using these measurement results and corresponding human judgment values on MC30to compute their Pearson correlation coefficient respectively. In addition, in order to increase the comprehensive of comparison, we compare also listed some algorithm results based on WordNetin table 1.

Table 1: Calculated Pearson correlation coefficient in different methods and MC30

Similarity method	Type	Semantic dictionary	Pearson correlation coefficient
Resnik	IC	WordNet	0.795
CP/CV	IC	WordNet	0.8138
Wu	Depth and path	WordNet	0.7464
Hao[6]	Depth and path	WordNet	0.8161
Liu	Depth and path	WordNet	0.8018
Mohamed[7]	Hybrid method	WordNet	0.8460
Qun Liu	Meaning Computation	HowNet	0.6991
Feng Li	Meaning Computation	HowNet	0.793
Proposed method	Depth and path	HowNet	0.8597

Analysis. The proposed method avoids the situation that similarity between the non-synonymous words is 1. In addition, computed similarity value of proposed method is closest to MC30 human judgment value compared to methods of Qun Liu and Feng Li. Furthermore proposed method is direct to measure word similarity, but need not to compute meaning similarity which requires measuring the similarity between sememe groups in the DEF of concept first. So the computational precision and efficiency of our proposed method is superior to that of Liu [2] and Li [5] .

Seen from table 1, proposed method has reached the excellent level of WordNet-based algorithms. But the proposed method also has some deficiencies. For example, similarity between “fruit” and “food” is always smaller, of which reason is that it takes six steps to search up their lowest common hypernym, which causes the shortest path length between two concepts too long. In addition, similarities between “monk” and “slave”, “lad” and “brother”, “wizard” and “lad” all are too large, of which reason is that their lowest common hypernym all are the first basic sememe in their DEF, which caused the shortest path length between two concepts too short.

Conclusions

This article puts forward word similarity computation method based on concept tree in HowNet [2] by constructing a concept tree and combining with a better method based on depth and path, which reasonably considers all kinds of sememes’ influences in the DEF and avoids the complex meaning similarity computation process that requires measuring the similarity between sememe groups in the DEF of concept. The experimental results show that proposed method is a better word similarity algorithm compared to other methods based on HowNet. In the research process of this method, we find that the computed similarities between some words may restrain each other by using the methods based on the tree. We think only by the continuous extension and updating of world knowledge base and constant improvement of lexicon architecture, this problem can be solved completely.

Acknowledgements

This work was financially supported by the National Natural Science Foundation of China (61363036 and 61462010), Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing.

References

[1] Information on http://www.keenage.com

[2] X.Y. Liu, Y.M. Zhou, R..S. Zheng, Measuring semantic similarity in WordNet, In: Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, Hong Kong, 3431–3435(2007).

[3] T. Berners-Lee, J. Hendler, O. Lassila, Scientific American, Vol. 284(5)(2001), pp. 34-43.

[4] F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider: The Description Logic Handbook: Theory, Implementation and Applications(Cambridge University Press, Cambridge 2003).

[5] Feng Li, Fang Li, Journal of Chinese information, Forum Vol. 03(2007), pp. 99-105, In Chinese.

[6] Dou Hao, Wanli Zuo, Tao Peng, Fengling He, An approach for calculating semantic similarity between words using WordNet, In: ICDMA, 177–180(2011).

[7] A.H.T. Mohamed, B.A. Mohamed, A.B. Hamadou, Journal of Engineering Applications of Artificial Intelligence, Vol. 36(2014), 238-261.