A SEMANTIC SIMILARITY MEASURE BETWEEN SENTENCES

Manh Hung Nguyen
Posts and Telecommunications Institute of Technology

Dinh Que Tran
Posts and Telecommunications Institute of Technology

Southeast-Asian Journal of Sciences, Vol 3, No 1 (2014), pp. 63-75

A SEMANTIC SIMILARITY MEASURE BETWEEN SENTENCES

Manh Hung Nguyen[1,2] and **Dinh Que Tran**[1]

[1] *Post and Telecommunication Institute of Technology (PTIT) Hanoi, Vietnam*

[2] *UMI UMMISCO 209 (IRD/UPMC), Hanoi, Vietnam*

e-mail: nmhufng@yahoo.com, tdque@yahoo.com

Abstract

The purpose of this paper is to present a mathematical model for estimating semantic similarity among sentences in texts. The similarity measure is constructed from the semantic similarity among concepts and a set of concepts. Based on this model, we develop algorithms to calculate the semantic similarity between two set of concepts and then the ones to estimate the semantic similarity between sentences. This work is considered as a continuation of our research [18] on the model of semantic similar measures among sentences.

1. Introduction

Semantic similarity, which is the form of semantic relatedness, has become one of important research areas in computation. It has been widely used in applications including natural language processing, document comparison, artificial intelligence, semantic web, semantic web service and semantic search engines. In the context of sentences, Jiang and Conrath [8] presented an approach for measuring semantic similarity/distance between words and concepts. It combines a lexical taxonomy structure with corpus statistical information. Lin [10] whose idea is to measure the similarity between any two objects based

Key words: mathematical model, semantic similarity, semantic matching, ontology, sentence similarity.
2000 AMS Mathematics Subject Classification: Applied Mathematics

on information-theoretic approach. Turney [19] introduced Latent Relational Analysis (LRA), a method for measuring semantic similarity based on the semantic relations between two pairs of words. Li et al. [9] presented an algorithm that takes account of semantic information and word order information implied in the sentences. The semantic similarity of two sentences is calculated using information from a structured lexical database and from corpus statistics.

Mihalcea et al. [4, 13] presented a method for measuring the semantic similarity of texts, using corpus-based and knowledge-based measures of similarity. Hliaoutakis et al. [6] investigated approaches to computing the semantic similarity between natural language terms (using WordNet as the underlying reference ontology) and between medical terms (using the MeSH ontology of medical and biomedical terms). Islam and Inkpen [7] presented a method for measuring the semantic similarity of texts using a corpus-based measure of semantic word similarity and a normalized and modified version of the Longest Common Subsequence (LCS) string matching algorithm. Ramage et al. [17] proposed an algorithm which aggregates local relatedness information via a random walk over a graph constructed from an underlying lexical resource such as Wordnet. Gad and Kamel [5] proposed a semantic similarity based model (SSBM). The semantic similarity based model computes semantic similarities by utilizing WordNet as an ontology. Madylova and Oguducu [12] presented a method for calculating semantic similarities between documents. This method is based on cosine similarity calculation between concept vectors of documents obtained from a taxonomy of words that captures IS-A relations. Castillo and Cardenas [3] presented a Recognizing Textual Entailment system which uses semantic distances to sentence level over WordNet to assess the impact on predicting Textual Entailment datasets. Pedersen [16] presented an empirical comparison of similarity measures for pairs of concepts based on Information Content. Oliva et al. [15] presented SyMSS, a method for computing short-text and sentence semantic similarity. The method is based on the notion that the meaning of a sentence is made up of not only the meanings of its individual words, but also the structural way the words are combined. Batet et al. [1] proposed a measure based on the exploitation of the taxonomical structure of a biomedical ontology. Bollegala et al. [2] proposed an empirical method to estimate semantic similarity using page counts and text snippets retrieved from a web search engine for two words. Lintean and Rus [11] proposed word-to-word semantic similarity metrics to quantify the semantic similarity at sentence level.

Tran and Nguyen [18] proposed a model to measure the semantic similarity between concepts and sets (non ordered) of concepts. Novelli and Oliveira [14] presented TextSSimily, a method that compares documents semantically considering only short text for comparison (text summary). Saric et al. [20] described the two systems for determining the semantic similarity of short texts using a support vector regression model with multiple features measuring word-overlap similarity and syntax similarity.

In this paper, we introduce a mathematical model for semantic similarity estimation in domains with various ontologies. First of all, we investigate a mathematical representation of semantic distance between concepts in an ontology. Then, we examine a mathematical model for similarity of two concepts as well as similarity between sentences. The significance of the proposed mathematical model is that it offers a generalization that enables to maintain flexibility and thus supports various computational measures. The remainder of this paper is organized as follows. Section presents our mathematical model for semantic similarity measure between two words. Section presents our mathematical model for semantic similarity measure between two sentences. The final section is conclusion and perspectives.

2. Backgrounds

2.1. Semantic Similarity between Concepts in an Ontology

Definition 1. *([18]) An ontology is a 2-tuple $\mathcal{G} =< \mathcal{C}, \mathcal{V} >$, in which $\mathcal{C}$ is a set of nodes corresponding to concepts defined in the ontology and $\mathcal{V}$ is a set of arcs representing relationships of couples of nodes in $\mathcal{C}$.*

Definition 2. *([18]) Let $\mathcal{C}$ be a set of concepts. A similarity measure $sim : \mathcal{C} \times \mathcal{C} \to [0,1]$ is a function from a pair of concepts to a real number between zero and one such that:*

(i) $\forall x \in \mathcal{C}\ sim(x,x) = 1$;

(ii) $\forall x, y \in \mathcal{C}\ sim(x,y) = sim(y,x)$.

Definition 3. *([18]) The path length $L(c_1, c_2)$ between concepts c_1 and c_2 in an ontology is the length of the shortest path from node c_1 to node c_2 on the ontology.*

Let c_0 be the nearest common ancestor concept of two concepts c_1 and c_2, we have $L(c_1, c_2) = L(c_1, c_0) + L(c_0, c_2)$. The semantic similar measure between c_1 and c_2 is based on the pre-similar function defined as follows:

Definition 4. *([18]) A function $f : \Re \times \Re \to [0,1]$ is pre-similar, denoted pre-sim, iff it satisfies the following conditions:*

(i) $f(0,0) = 1$;

(ii) $f(\infty, l) = f(l, \infty) = 0$;

(iii) $f(l_1, l_2) = f(l_2, l_1)$;

(iv) $f(l_1, l_2) \geqslant f(l_3, l_4)$ *if* $l_1 + l_2 \leqslant l_3 + l_4$;

(v) $f(l_1, l_0) \geqslant f(l_2, l_0)$ *if* $l_1 \leqslant l_2$*;*

(vi) $f(l_0, l_1) \geqslant f(l_0, l_2)$ *if* $l_1 \leqslant l_2$*.*

Proposition 1. *([18]) Given a pre-sim function* $f_{ont} : \Re \times \Re \to [0, 1]$*. The function* $s_{ont} : \mathcal{C} \times \mathcal{C} \to [0, 1]$ *between concepts* c_1 *and* c_2 *with the nearest common ancestor* c_0 *on an ontology determined by the formula*

$$s_{ont}(c_1, c_2) = f_{ont}(L(c_1, c_0), L(c_0, c_2))$$

is a similar measure.

2.2. Syntax Similarity between Words with the Same Core

In reality, there are many of words with the same original core wordthat are not included in an ontology. In order to measure the semantic similarity between these words (called the core semantic similarity), we need an additional concept.

Definition 5. *([18]) The syntax distance between a word* w_1 *and its original core word* w_0*, denoted as* $d(w_1, w_0)$*, is the total number of characters that may be added (or deleted) from the word* w_1 *to become the original core word* w_0*.*

As a consequence, the syntax distance between two words w_1 and w_2, which have the same original core word $w_0 \notin \{w_1, w_2\}$, is the total distance from each of them to the common core word: $d(w_1, w_2) = d(w_1, w_0) + d(w_2, w_0)$. Let w_0 be the original core word of two words w_1 and w_2, we define a syntax similarity between w_1 and w_2 as follows:

Proposition 2. *([18]) Let* $f_{syn} : \Re \times \Re \to [0, 1]$ *be a pre-similar function. The syntax similarity between words* w_1 *and* w_2 *determined by the formula*

$$s_{syn}(w_1, w_2) = f_{syn}(d(w_1, w_0), d(w_2, w_0))$$

is a similar measure.

2.3. Transitive Semantic Similarity

Let c_1, c_2 and c_3 be concepts, in which only c_2 and c_3 belong to the same ontology and c_1 and c_2 shares the same core word. Then the relatedness relation between c_1 and c_3 is called a *transitive semantic relation.*

Definition 6. *([18]) A function* $f_{tran} : \Re \times \Re \to [0, 1]$ *is a transitive similar function, denoted tra-sim, iff it satisfies the following conditions:*

(i) $0 \leqslant f_{tran}(u, v) \leqslant v$*;*

(ii) $f_{tran}(u_1, v) \leqslant f_{tran}(u_2, v)$ *if* $u_1 \leqslant u_2$*;*

(iii) $f_{tran}(u, v_1) \leqslant f_{tran}(u, v_2)$ *if* $v_1 \leqslant v_2$.

And the transitive semantic distance is defined as follows:

Definition 7. *([18]) Let* c_1*,* c_2 *and* c_3 *be concepts, in which only* c_2 *and* c_3 *belong to the same ontology and* c_1 *and* c_2 *shares the same core word. Suppose that* $f_{tran} : \Re \times \Re \to [0, 1]$ *is a tra-sim function,* $s_{syn}(c_1, c_2)$ *is the syntax similarity on the same core word between* c_1 *and* c_2*,* $s_{ont}(c_2, c_3)$ *is the semantic similarity on ontology between* c_2 *and* c_3*. The transitive semantic similarity between concepts* c_1 *and* c_3 *via concept* c_2 *is determined by the following formula:*

$$s_{tran}(c_1, c_2, c_3) = f_{tran}(s_{syn}(c_1, c_2), s_{ont}(c_2, c_3))$$

It is easy to prove the following proposition.

Proposition 3. *([18]) Suppose that* c_1 *has many concepts in core word relations* $C = \{c'_1, c'_2, ..., c'_n\}$ *and all* $c'_i \in C$ *have semantic similarity on an ontology with* c_3*. The transitive semantic similarity between* c_1 *and* c_3 *defined by the following formula:*

$$s_{tran}(c_1, c_3) = \quad Max_{c'_i \in C}\{f_{tran}(s_{syn}(c_1, c'_i), s_{ont}(c'_i, c_3))\} \tag{1}$$

is a similar measure.

2.4. General Semantic Similarity between Two Concepts

Let c_1 and c_2 be two words or concepts. We consider the following cases:

- If c_1 and c_2 are both in the same ontology, then their general semantic similarity is their ontology-based semantic similarity defined in Definition 4;
- If either c_1 or c_2 is in an ontology, other is not, their general semantic similarity is their transitive semantic similarity defined in Definition 7;
- If neither c_1 nor c_2 is in an ontology, we consider as they have not any semantic relation;

Definition 8. *([18]) Given* c_1 *and* c_2 *be the two words or concepts, the semantic similarity between them is determined by the formula:*

$$s_{word}(c_1, c_2) = \begin{cases} s_{ont}(c_1, c_2) \ \textit{if } c_1, c_2 \in \textit{ an ontology} \\ s_{tran}(c_1, c_2) \ \textit{if } c_1 \textit{ or }_2 \in \textit{ an ontology} \\ s_{syn}(c_1, c_2) \ \textit{if } c_1, c_2 \notin \textit{ any ontology} \end{cases}$$

where $s_{ont}(c_1, c_2)$ *is the semantic similarity based on ontology,* $s_{tran}(c_1, c_2)$ *is the transitive similarity, and* $s_{syn}(c_1, c_2)$ *is syntax similarity between* c_1 *and* c_2*.*

3. Semantic Similarity between Two Sentences

In this section, we consider a sentence as an ordered set of words. And then the similarity between two sentences (two sets of words) is examined with two levels:

- Semantic similarity: Only the semantic is considered, the order of word in the set is not considered.
- Order similarity: only the order of words in the set is considered, the semantic is not considered.

3.1. Semantic Similarity between Two Set of Words

Let $S_1 = \{c_1^1, c_1^2, ..., c_1^m\}$ and $S_2 = \{c_2^1, c_2^2, ..., c_2^n\}$ be the two considered sets of words, we create a *common set* of these two sets $S_{12} = S_1 + S_2 = \{c^1, c^2, ..., c^{m+n}\}$. And then construct the two corresponding non-ordered semantic vectors $T_i = (t_i^1, t_i^2, ..., t_i^{m+n}), i = 1, 2$ as:

$$t_i^j = \begin{cases} 1 \text{ if } c^j \in S_i \\ max\{s_{word}(c^j, c_i^v)\}, v = 1, n \text{ or } m \text{ if } c^j \notin S_i \end{cases}$$

where $s_{word}(c^j, c_i^v)$ is the semantic similarity between the two words c^j and $c_i^v \in S_i, i = 1, 2$.

In order to measure the semantic similarity between two non-ordered sets of words S_1 and S_2, we make use of the following assumptions:

Assumption 1. *Let T_1 and T_2 be the two non-ordered semantic vectors of S_1 and S_2:*

- *The bigger the magnitude of each vector $T_i, i = 1, 2$ is, the higher the semantic similarity between S_1 and S_2 is.*

Definition 9. *A function $f_{noss} : [0, 1]^k \times [0, 1]^k \rightarrow [0, 1]$ is a semantic similar function between two non-ordered sets of words, denoted* Non-Ordered-Set-Similarity (NOSS)*, if it satisfies the following conditions:*

(i) $f_{noss}((0, 0, ..., 0), (0, 0, ..., 0)) = 0$;

(ii) $f_{noss}((1, 1, ..., 1), (1, 1, ..., 1)) = 1$;

(iii) $f_{noss}(X_1, Y) \leqslant f_{noss}(X_2, Y)$ *if* $\|X_1\| \leqslant \|X_2\|$

(iv) $f_{noss}(X, Y_1) \leqslant f_{noss}(X, Y_2)$ *if* $\|Y_1\| \leqslant \|Y_2\|$

Proposition 4. *The following functions are* Non-Ordered-Set-Similarity (NOSS) *functions:*

(i) $f((x_1, x_2, ..., x_n), (y_1, y_2, ..., y_n)) = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{x_i + y_i}{2} \right)$

(ii) $f((x_1, x_2, ..., x_n), (y_1, y_2, ..., y_n)) = \sqrt{\frac{\sum_{i=1}^{n} \left(\frac{x_i + y_i}{2} \right)^2}{n}}$

(iii) $f((x_1, x_2, ..., x_n), (y_1, y_2, ..., y_n)) = \frac{2\sqrt{\sum_{i=1}^{n} x_i^2} * \sqrt{\sum_{i=1}^{n} y_i^2}}{n * \sqrt{\sum_{i=1}^{n} (x_i + y_i)^2}}$

And the semantic similarity between two non-ordered sets of words S_1 and S_2 is defined as follows:

Definition 10. *Given $S_1 = \{c_1^1, c_1^2, ..., c_1^m\}$ and $S_2 = \{c_2^1, c_2^2, ..., c_2^n\}$ be the two considered sets of words, and let $T_1 = (t_1^1, t_1^2, ..., t_1^{m+n})$ and $T_2 = (t_2^1, t_2^2, ..., t_2^{m+n})$ be the semantic vector of S_1 in comparing with S_2 and that of S_2 in comparing with S_1, respectively. The semantic similarity between two non-ordered sets of words S_1 and S_2 is determined by the formula:*

$$s_{noss}(S_1, S_2) = f_{noss}(T_1, T_2).$$

where $f_{noss}(x, y)$ is a Non-Ordered-Set-Similarity (NOSS) function.

The algorithm of estimating the semantic similarity between two sets of concepts S_1 and S_2 is presented in Algorithm 1. We firstly calculate the *common set* of S_1 and S_2 (Step 1), then initiate and construct the two vector T_1 and T_2 (Step 2-13), and then calculate the semantic similarity of the two sets by f_{noss} function (Step 14).

3.2. Order Similarity between Two Sets of Words

Let $S_1 = \{c_1^1, c_1^2, ..., c_1^m\}$ and $S_2 = \{c_2^1, c_2^2, ..., c_2^n\}$ be the two considered sets of words, we also create a *minimal common set* of these two sets $S_{12} = S_1 \cup S_2 = \{c^1, c^2, ..., c^k\}$. And then construct the two corresponding ordered vectors $T_i = (t_i^1, t_i^2, ..., t_i^k), i = 1, 2$ as follows:

$$t_i^j = \begin{cases} l \text{ if } c^j = c_i^l \in S_i \\ 0 \text{ if } c^j \notin S_i \end{cases}$$

In order to measure the order similarity between two sets of words S_1 and S_2, we make use of the following assumptions:

Assumption 2. *Let T_1 and T_2 be the two ordered vectors of S_1 and S_2:*

- *The order similarity between S_1 and S_2 is highest when the two vectors T_1 and T_2 are identical and there is no element of value 0.*

Algorithm 1 Semantic similarity between two sets of concepts

Input: 2 sets of words $S_1 = \{c_1^1, c_1^2, ..., c_1^m\}$ and $S_2 = \{c_2^1, c_2^2, ..., c_2^n\}$
Output: the semantic similarity between S_1 and S_2: $SemSetSim(S_1, S_2)$

$S_{12} \leftarrow S_1 + S_2$
$T_1 \leftarrow (0, ..., 0)$
$T_2 \leftarrow (0, ..., 0)$
for all t_i in the T_1 **do**
    for all c_j in the S_1 **do**
        $t_i \leftarrow max\{s_{word}(c_i, c_j)\}$
    end for
end for
for all t_i in the T_2 **do**
    for all c_j in the S_2 **do**
        $t_i \leftarrow max\{s_{word}(c_i, c_j)\}$
    end for
end for
$SemSetSim(S_1, S_2) \leftarrow f_{noss}(T_1, T_2)$
return$SemSetSim(S_1, S_2)$

- *The more the vector T_1 is similar to vector T_2, the higher the order similarity between S_1 and S_2 is.*

Definition 11. *A function $f_{oss} : \Re^n \to [0, 1]$ is a semantic similar function between two ordered sets of words, denoted* Ordered-Set-Similarity (OSS)*, if it satisfies the following conditions:*

(i) $f_{oss}(0, 0...0) = 1$*;*

(ii) $f_{oss}(x_1, x_2, ...x_n) \leqslant f_{oss}(y_1, y_2, ...y_n)$ *if* $x_i \geqslant y_i$ *with* $\forall i = 1, n$

Proposition 5. *The following functions are* Ordered-Set-Similarity (OSS) *functions:*

(i) $f_{oss}(x_1, x_2, ..., x_n) = 1 - \dfrac{\sqrt{\sum_{i=1}^n x_i^2}}{n}$

(ii) $f_{oss}(x_1, x_2, ..., x_n) = 1 - \dfrac{\sum_{i=1}^n x_i}{n}$

And the order similarity between two ordered sets of words S_1 and S_2 is defined as follows:

Definition 12. *Given $S_1 = \{s_1^1, s_1^2, ..., s_1^m\}$ and $S_2 = \{s_2^1, s_2^2, ..., s_2^n\}$ be the two considered sets of words, and let $T_1 = (t_1^1, t_1^2, ..., t_1^{m+n})$ and $T_2 = (t_2^1, t_2^2, ..., t_2^{m+n})$*

be the order vector of S_1 *in comparing with* S_2 *and that of* S_2 *in comparing with* S_1*, respectively. The order similarity between two ordered sets of words* S_1 *and* S_2 *is determined by the formula:*

$$s_{oss}(S_1, S_2) = f_{oss}(d_1, d_2, ...d_{m+n}).$$

where:

$$d_i = \begin{cases} \dfrac{\mid t_1^i - t_2^i \mid}{max(m,n)} \; if \; min(t_1^i, t_2^i) \neq 0 \\ 1 \; if \; min(t_1^i, t_2^i) = 0 \end{cases}$$

and $f_{oss}(d_1, d_2, ...d_{m+n})$ *is an Ordered-Set-Similarity (OSS) function.*

The algorithm of estimating the order similarity between two sets of concepts S_1 and S_2 is presented in Algorithm 2. First, constructing the *minimal common set* of two given sets (Step 1). Then we initiate and construct the two vector T_1 and T_2 (Steps 2-17). And then we calculate the distance vector between T_1 and T_2 (Steps 18-24). Lastly, applying the f_{oss} function to calculate the order similarity of two given sets (Step 25).

3.3. Similarity between Two Sentences

Let S_1 and S_2 be the two considered sentences, it means that they are two ordered sets of words. Let also $S_{noss}(S_1, S_2)$ and $S_{oss}(S_1, S_2)$ be respectively the semantic similarity and the order similarity between S_1 and S_2. In order to measure the semantic similarity between two sentences S_1 and S_2, we make use of the following assumptions:

Assumption 3. *Let* $S_{noss}(S_1, S_2)$ *and* $S_{oss}(S_1, S_2)$ *be respectively the semantic similarity and the order similarity between* S_1 *and* S_2*:*

- *The higher the semantic similarity* $S_{noss}(S_1, S_2)$ *is, the higher the semantic similarity between* S_1 *and* S_2 *is, and vice versa.*
- *The higher the order similarity* $S_{oss}(S_1, S_2)$ *is, the higher the semantic similarity between* S_1 *and* S_2 *is, and vice versa.*

Definition 13. *A function* $f : [0,1] \times [0,1] \rightarrow [0,1]$ *is a semantic and order similar function between two objects of sentences, denoted* Semantic-and-Order-Similarity (SOS)*, if it satisfies the following conditions:*

(i) $f_{sos}(0, 0) = 0$*;*

(ii) $f_{sos}(1, 1) = 1$*;*

(iii) $f_{sos}(x_1, y) \leqslant f_{sos}(x_2, y)$ *if* $x_1 \leqslant x_2$

Algorithm 2 Order similarity between two sets of concepts

Input: 2 sets of words $S_1 = \{c_1^1, c_1^2, ..., c_1^m\}$ and $S_2 = \{c_2^1, c_2^2, ..., c_2^n\}$
Output: the order similarity between S_1 and S_2: $OrdSetSim(S_1, S_2)$

```
S_12 ← S_1 ∪ S_2
T_1 ← (0, ..., 0)
T_2 ← (0, ..., 0)
for all c_i in the S_12, t_i in the T_1 do
    if c_i in the S_1 then
        t_i ← 1
    else
        t_i ← 0
    end if
end for
for all c_i in the S_12, t_i in the T_2 do
    if c_i in the S_2 then
        t_i ← 1
    else
        t_i ← 0
    end if
end for
for all t_i^1 in the T_1, t_i^2 in the T_2,  do
    if min(t_i^1, t_i^2) ≠ 0  then
        d_i ← |t_1^i − t_2^i| / max(m,n)
    else
        d_i ← 1
    end if
end for
OrdSetSim(S_1, S_2) ← f_oss(d_1, d_2, ...d_{m+n})
returnOrdSetSim(S_1, S_2)
```

(iv) $f_{sos}(x, y_1) \leqslant f_{sos}(x, y_2)$ *if* $y_1 \leqslant y_2$

Proposition 6. *The following functions are* Semantic-and-Order-Similarity (SOS) *functions:*

(i) $f(x, y) = x * y$

(ii) $f(x, y) = w_1 * x + w_2 * y, \forall w_1, w_2 \in [0, 1], w_1 + w_2 = 1$

And the semantic similarity between two sentences S_1 and S_2 is defined as follows:

Definition 14. *Given S_1 and S_2 be the two considered objects of sentences, it means that they are two ordered sets of words. Let also $S_{noss}(S_1, S_2)$ and $S_{oss}(S_1, S_2)$ be respectively the semantic similarity and the order similarity between S_1 and S_2. The semantic similarity between two sentences S_1 and S_2 is determined by the formula:*

$$S_{os}(S_1, S_2) = f_{sos}(s_{noss}(S_1, S_2), s_{oss}(S_1, S_2)).$$

where $f_{sos}(x, y)$ is an Semantic-and-Order-Similarity (SOS) function.

The algorithm of estimating the semantic similarity between two sentences S_1 and S_2 is presented in Algorithm 3. We firstly calculate the *semantic similarity* of the two non-ordered sets of words S_1 and S_2 (Step 1), then calculate the *order similarity* of the two ordered sets of words S_1 and S_2 (Step 2), and then calculate the semantic similarity of the two sentences by f_{sos} function (Step 3).

Algorithm 3 Semantic similarity between two sentences

Input: 2 sentences S_1 and S_2
Output: the semantic similarity between S_1 and S_2: $SenSim(S_1, S_2)$

1: $x \leftarrow s_{noss}(S_1, S_2)$
2: $y \leftarrow s_{oss}(S_1, S_2)$
3: $SenSim(S_1, S_2) \leftarrow f_{sos}(x, y)$
return$SenSim(S_1, S_2)$

4. Conclusions

In this paper, we presented a mathematical model for calculating the semantic similarity between sentences. We first estimate the semantic similarity between two concepts which are either defined in an ontology, or only one of them is defined in an ontology. The estimation is based on their semantic relation on ontology, or their syntax relation or both of them. And then, the semantic similarity between two sentences is constructed on the semantic similarity between the individual words of them. Our model is considered as a generalization of the proposed similarity computational models. At each step of estimation, instead of applying a particular function, we generate them as some series of functions satisfying the constraints defined by the model. This makes our model more flexible in developing. It means that the developers could choose their own operators and functions from their special domain as long as they satisfy the constraints defined in our approach. The semantic similarity of two texts with several sentences will be considered and presented in our future work.

References

[1] Montserrat Batet, David Sánchez, and Aida Valls. An ontology-based measure to compute semantic similarity in biomedicine. *J. of Biomedical Informatics*, 44(1):118–125, February 2011.

[2] Danushka Bollegala, Yutaka Matsuo, and Mitsuru Ishizuka. A web search engine-based approach to measure semantic similarity between words. *IEEE Trans. on Knowl. and Data Eng.*, 23(7):977–990, July 2011.

[3] Julio J. Castillo and Marina E. Cardenas. Using sentence semantic similarity based on wordnet in recognizing textual entailment. In *Proceedings of the 12th Ibero-American Conference on Advances in Artificial Intelligence*, IBERAMIA'10, pages 366–375, Berlin, Heidelberg, 2010. Springer-Verlag.

[4] Courtney Corley and Rada Mihalcea. Measuring the semantic similarity of texts. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, EMSEE '05, pages 13–18, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[5] WalaaK. Gad and MohamedS. Kamel. New semantic similarity based model for text clustering using extended gloss overlaps. In Petra Perner, editor, *Machine Learning and Data Mining in Pattern Recognition*, volume 5632 of *Lecture Notes in Computer Science*, pages 663–677. Springer Berlin Heidelberg, 2009.

[6] Angelos Hliaoutakis, Giannis Varelas, Epimeneidis Voutsakis, Euripides G. M. Petrakis, and Evangelos Milios. Information retrieval by semantic similarity. In *Intern. Journal on Semantic Web and Information Systems (IJSWIS), 3(3):5573, July/Sept. 2006. Special Issue of Multimedia Semantics*, 2006.

[7] Aminul Islam and Diana Inkpen. Semantic text similarity using corpus-based word similarity and string similarity. *ACM Trans. Knowl. Discov. Data*, 2(2):1–25, July 2008.

[8] Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. *CoRR*, cmp-lg/9709008, 1997.

[9] Yuhua Li, David McLean, Zuhair A. Bandar, James D. O'Shea, and Keeley Crockett. Sentence similarity based on semantic nets and corpus statistics. *IEEE Trans. on Knowl. and Data Eng.*, 18(8):1138–1150, August 2006.

[10] Dekang Lin. An information-theoretic definition of similarity. In *Proc. 15th International Conf. on Machine Learning*, pages 296–304. Morgan Kaufmann, San Francisco, CA, 1998.

[11] Mihai C. Lintean and Vasile Rus. Measuring semantic similarity in short texts through greedy pairing and word semantics. In G. Michael Youngblood and Philip M. McCarthy, editors, *Proceedings of the Twenty-Fifth International Florida Artificial Intelligence Research Society Conference, Marco Island, Florida. May 23-25, 2012*. AAAI Press, 2012.

[12] A. Madylova and S.G. Oguducu. A taxonomy based semantic similarity of documents using the cosine measure. In *Computer and Information Sciences, 2009. ISCIS 2009. 24th International Symposium on*, pages 129–134, Sept 2009.

[13] Rada Mihalcea, Courtney Corley, and Carlo Strapparava. Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 1*, AAAI'06, pages 775–780. AAAI Press, 2006.

[14] Andreia Dal Ponte Novelli and Jose Maria Parente De Oliveira. Article: A method for measuring semantic similarity of documents. *International Journal of Computer Applications*, 60(7):17–22, December 2012.

[15] Jess Oliva, Jos Ignacio Serrano, Mara Dolores del Castillo, and ngel Iglesias. Symss: A syntax-based measure for short-text semantic similarity. *Data & Knowledge Engineering*, 70(4):390–405, 2011.

[16] Ted Pedersen. Information content measures of semantic similarity perform better without sense-tagged text. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 329–332, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[17] Daniel Ramage, Anna N. Rafferty, and Christopher D. Manning. Random walks for text semantic similarity. In *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing*, TextGraphs-4, pages 23–31, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[18] Dinh Que Tran and Manh Hung Nguyen. A mathematical model for semantic similarity measures. *South-East Asian Journal of Sciences*, 1(1):32–45, 2012.

[19] Peter D. Turney. Measuring semantic similarity by latent relational analysis. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, IJCAI'05, pages 1136–1141, San Francisco, CA, USA, 2005. Morgan Kaufmann Publishers Inc.

[20] Frane Šarić, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. Takelab: Systems for measuring semantic text similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, pages 441–448, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.