Towards Automatic Generation of "Preference Profile" for Primitive Concept Similarity Measures on SNOMED CT

Htet Htet Htun[1]
School of ICT, Sirindhorn International Institute of Technology, Thammasat University, Thailand
Email: htethtethtun.8910@gmail.com

Virach Sornlertlamvanich[2]
School of ICT, Sirindhorn International Institute of Technology, Thammasat University, Thailand
Email: virach@siit.tu.ac.th

Boontawee Suntisrivaraporn[3]
School of ICT, Sirindhorn International Institute of Technology, Thammasat University.
Customer Insights Department, Marketing Group, dtac, Thailand
Email: boontawee.suntisrivaraporn@dtac.co.th

***Abstract*—Recently, a non-standard reasoning service of measuring similarity between two concepts has been proposed for Description Logic (DL) ontologies, in addition to the classical reasoning service of testing subsumption and logical equivalence. One of the previous works suggests that similarity not only depends on the objective aspects (i.e. concept descriptions of the two concepts), but is also influenced by the subjective factors (i.e. judgments of the viewing agent). In this paper, we propose to employ various text similarity measures to compare the textual annotations of primitive concepts as well as primitive roles from the side of estimating human experts' interpretations. A collection of primitive similarity degrees obtained in this way is regarded as an automatically-generated possible doctors' judgments (preference profile) for primitive similarity measures. We perform extensive experiments on the renown clinical ontology SNOMED CT. After generating the primitive concepts similarity measures with various similarity methods, this paper presented interesting findings from the experiments and discuss benefits and usability of our approach.**

Keywords — **Concept Similarity Measures, Text Similarity, Preference Profile, SNOMED CT, Description Logic**

I. Introduction

Concept similarity measure (CSM) is one of the non-standard Description Logic (DL) reasoning services. It determines how similar two concepts are and returns the numerical value between 0 and 1 that represents their degree of similarity. In DL, similarity measures between two concept descriptions have been developed. For example, there are two concept descriptions from SNOMED CT ontology as the following.

Hypoxia ≡ DisorderOfRespiratorySystem ⊓ DisorderOf BloodGas ⊓ ∃ roleGroup.(∃ interprets. OxygenDelivery)

Hypoxemia ≡ DisorderOfRespiratorySystem ⊓ DisorderOf BloodGas ⊓ ∃ roleGroup.(∃ findingSite. ArterialSystemStructure)

For the above two concept descriptions, many semantic similarity measures using the hierarchy of concepts have been developed such as ELSIM(semantic similarity reasoner) [2]. In the SNOMED CT ontology, there has concept descriptions for defined concept names and others concept names called primitive concepts that have no descriptions. After developing the similarity between two concept descriptions for defined concept names, let consider the case of similarity between primitive concepts. Because the role of primitive concept similarity is also important in realistic similarity measures, eg: medical treatment cases. When medical experts want to find an appropriate treatment for the current patient based on previous treatments in hospital database, they have to consider the similarity of characteristics (primitive names) between the previous and current treatments. Therefore, this paper generate primitive concepts similarity measures between the pair of primitive names on SNOMED CT which is the terminology that includes medical information for patient care.

In the SNOMED CT, there are about 364,461 concepts that includes primitive concepts and primitive roles names that show medical terms from different top level categories. For finding the similarity of primitive concepts, if we want to compare the similarity as logic representations (concept descriptions), we have to define primitive concept names as concept descriptions. However, writing the concept name into logic representation is too difficult all over the world. For example, if we compare the two primitive concepts "structure of capsule of kidney"and "structure of capsule of red nucleus" as logic descriptions, we have to identify the attributeName = "structure", attributeValue= "capsule" and we must transform as the following.

structureOfCapsuleOfKidney ≡ kidney ⊓ ∃ hasStructure.capsule

structureOfCapsuleOfRedNucleus ≡ red nucleus ⊓ ∃ hasStructure.capsule

Moreover, it takes too long time to identify each concept name into logic descriptions. And we also need checking process from domain experts to ensure that these converted descriptions are correct or not. Therefore, we don't think the similarity measures of primitive concepts from the side of computing as logic representations. We think the similarity measures of primitive concepts from the side of how the human experts interpreted on the primitive concepts. For the

medical concepts, doctors are human experts. Therefore, we will consider the primitive concept similarity as the way of how the doctors understand the similarity between primitive concepts.

When doctors consider the similarity of primitive concepts, they can have different interpretations between the concepts. For medical treatment cases, doctors can set the similarity of two primitive concepts (eg: "structure of kidney" and "structure of red nucleus") as 0.3 or 0.5 or some values because they can think that these two concepts are different parts of human body but they have common meaning (structure) according to the terms in concept names. For two others primitive concepts "structure of capsule of kidney" and "tumor-like lesion of skin", doctors can set similarity degree is zero because these two are totally different human parts and there is no common meaning (common terms). Therefore, doctors generally understand the similarity of primitive concepts by the terms expressed in concept names but different doctors can also have a little different similarity values based on their judgments. So, the similarity degree of the pair of primitive concepts depends on how the doctors conceive by the terms.

From this point of view, this paper propose to employ various text similarity methods to get different doctors' judgments automatically on primitive concepts and primitive roles on SNOMED CT. For example, we will generate different similarity values for two primitive concepts "structure of kidney" and "structure of red nucleus" by employing various textual similarity methods because doctors also consider the similarity of primitive concepts based on equality of terms in concept names. Since we think the primitive concepts similarity as the way of human experts understanding, similarity values obtained from each textual method can be regarded as automatically-generated possible doctors' judgments (preference profile) for primitive concepts and primitive roles. Definitions of primitive concept and primitive role similarity are as follows.

Primitive concept similarity : Let $\mathsf{CN}^{\mathsf{pri}}(\mathcal{T})$ be a set of primitive concept names occurring in terminology $\mathcal{T}$. A primitive concept similarity is a partial function [1] s^c: $\mathsf{CN} \times \mathsf{CN} \rightarrow$ [0,1], where $\mathsf{CN} \subseteq \mathsf{CN}^{\mathsf{pri}}(\mathcal{T})$. For two concept names $\mathsf{A},\mathsf{B} \in \mathsf{CN}^{\mathsf{pri}}(\mathcal{T})$, s^c (A,B) = s^c (B,A) and s^c(A,A) =1.

Primitive role similarity : Let $\mathsf{RN}^{\mathsf{pri}}(\mathcal{T})$ be a set of primitive role names occurring in a terminology $\mathcal{T}$ [1]. A primitive role similarity is a partial function s^r: RN $\times$ RN $\rightarrow$ [0,1], where RN $\subseteq$ $\mathsf{RN}^{\mathsf{pri}}(\mathcal{T})$. For two role names $r,s \in \mathsf{RN}^{\mathsf{pri}}(\mathcal{T})$, such that $s^r(r,s) = s^r(s,r)$ and $s^r(r,r) = 1$.

The rest of the paper is organized in the following order. Section II reviews the background of the DL and SNOMED CT that we apply for the experiments. Section III presents different similarity methods to generate various primitive similarity measures and calculation of these methods based on concepts in Table I. Section IV explains the experimental results of various similarity methods and findings from the experiments on SNOMED CT. Finally, section V presents the conclusion and future work.

II. Preliminaries

In Description Logics (DLs), concept descriptions are inductively defined with the help a set of constructors, with a set of concept names CN and a set of role names RN. The set of concept descriptions for a specific DL $\mathcal{ELH}$ is denoted by Con($\mathcal{ELH}$) [2]. The set Con($\mathcal{ELH}$) can be defined as follow:

$$\mathsf{C}, \mathsf{D} \rightarrow \mathsf{A} \mid \mathsf{T} \mid \mathsf{C} \sqcap \mathsf{D} \mid \exists r.\mathsf{C}$$

where T denotes the top concept, C, D $\in$ Con($\mathcal{ELH}$), A is concept name and r is role name. In DL, concept names appearing on the left-hand side of a definition are called defined concept names (CN^{def}). Other concept names are called primitive concept names (CN^{pri}) [3]. Therefore, CN= $\mathsf{CN}^{pri} \cup \mathsf{CN}^{def}$.

In Figure 1, there are two primitive concept names p_i and p_j or two primitive role names r_i and r_j from SNOMED CT. In real medical treatments, each doctor will consider the similarity values of (p_i,p_j) and (r_i,r_j) based on the equality of the terms in concept names according to their understanding. Therefore, this paper employ different textual similarity methods in order to get these possible doctors' judgments for primitive concepts and primitive roles names.

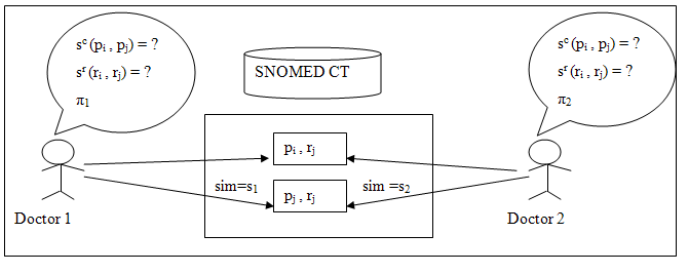

Fig. 1. Primitive similarity measures based on human interpretations

Let $\mathsf{CN}^{\mathsf{pri}}$ be the set of all primitive concept names and role names in SNOMED CT. For each $P \in \mathsf{CN}^{\mathsf{pri}}$, we denote by text(P), the textual annotation of P. For convenience, we denote by $tset(P)$, the set of words occurring in $text(P)$ as Table I.

TABLE I
PRIMITIVE CONCEPTS FROM SNOMED CT

Primitive concepts P	conceptID (P)	text(P)	tset(P)
P_1	254735005	"Tumor dermis"	{"Tumor","dermis"}
P_2	255103009	"malignant tumor mesothelial soft tissue"	{"malignant","tumor", "mesothelial","soft","tissue"}
P_3	254293002	"TNM tumor staging system"	{"TNM","tumor","staging", "system"}
P_4	106247005	"FIGO staging system epithelial tumor ovary"	{"FIGO","staging", "system", "epithelial","tumor","ovary"}

Analogously, tlist(P) denotes the lists of words occurring in $text(P)$.
tlist(P_1) = ["Tumor","dermis"]

tlist(P_2) = [“malignant”, “tumor”, “mesothelial”, “soft”, “tissue”]
tlist(P_3) = [“TNM”,“tumor”,“staging”,“system”]
tlist(P_4) = [“FIGO”, “staging”, “system”, “epithelial”, “tumor”, “ovary”]

III. Different Similarity Methods to Get Automatically-generated Preference Profiles

In this section, we describe different similarity methods to generate primitive concepts similarity degrees on SNOMED CT. We use two different kinds of methods, unordered-based and ordered-based methods as shown in Figure 2. The four concepts (P_1 - P_4) in Table I, are used as an example for explaining the similarity measures of each method.

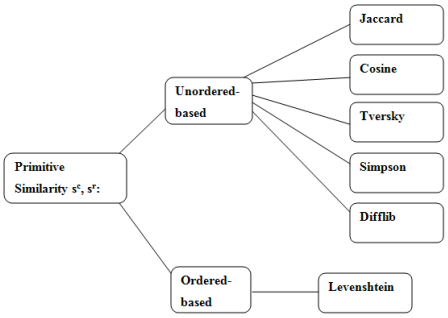

Fig. 2. Similarity methods to generate primitive similarity measures

A. Jaccard Similarity Coefficient

The Jaccard similarity coefficient is defined as the size of the intersection divided by the size of the union of two sets [4]. Equation 1 is defined for two concepts P_1 and P_2 in Table I.

$$\mathsf{tsim}^{Jaccard}(P_1, P_2) = \frac{\mid tset(P_1) \cap tset(P_2) \mid}{\mid tset(P_1) \cup tset(P_2) \mid} \quad (1)$$

According to Equation 1,
$\mathsf{tsim}^{Jaccard}(P_1,P_2)$ = 1/6 = 0.1667
$\mathsf{tsim}^{Jaccard}(P_1,P_3)$= 1/5 = 0.2000
$\mathsf{tsim}^{Jaccard}(P_3,P_4)$= 3/7 = 0.4286

B. Cosine Similarity Coefficient

Cosine similarity between two vectors [5] is measured by using the word vectors of a dot product and magnitude $||\cdot||$ as in Equation 2.

$$\mathsf{tsim}^{Cosine}(P_1, P_2) = \frac{tset(P_1).tset(P_2)}{\parallel tset(P_1) \parallel \parallel tset(P_2) \parallel} \quad (2)$$

TABLE II
Word Vector in sets

	tumor	dermis	malignant	mesothelial	soft	tissue
P_1	1	1	0	0	0	0
P_2	1	0	1	1	1	1

$\mathsf{tsim}^{Cosine}(P_1,P_2)$=

$$= \frac{(1*1+1*0+0*1+0*1+0*1+0*1)}{\sqrt{1^2+1^2+0+0+0+0}\sqrt{1^2+0+1^2+1^2+1^2+1^2}}$$
$$= 0.3162$$

$\mathsf{tsim}^{Cosine}(P_1,P_3)$= 0.3536 $\mathsf{tsim}^{Cosine}(P_3,P_4)$= 0.6124

C. Tversky Coefficient

In Tversky index [6] and [10], setting α, β = 1 produces the Jaccard coefficient and α, β = 0.5 produces Dice's coefficient. In our case, we used the value of α and β = 0.5.

$\mathsf{tsim}^{Tversky}(P_1,P_2)$=

$$\frac{\mid tset(P_1) \cap tset(P_2) \mid}{\mid tset(P_1) \cap tset(P_2) \mid + \alpha \mid tset(P_1) - tset(P_2) \mid + \beta \mid tset(P_2) - tset(P_1) \mid} \quad (3)$$

tset(P_1)- tset(P_2) means the set of elements in tset(P_1) but not in tset(P_2).

$\mathsf{tsim}^{Tversky}(P_1, P_2) = \frac{1}{1+(0.5*1)+(0.5*4)}$ = 0.2857
$\mathsf{tsim}^{Tversky}(P_1,P_3)$= 0.3333 $\mathsf{tsim}^{Tversky}(P_3,P_4)$= 0.6000

D. Szymkiewicz-Simpson Coefficient

Szymkiewicz-Simpson coefficient measures the overlap between two sets, and is defined as the ratio of the cardinality of the intersection to the minimum between the cardinality of the two sets [7].

$$\mathsf{tsim}^{Simpson}(P_1, P_2) = \frac{\mid tset(P_1) \cap tset(P_2) \mid}{min(\mid tset(P_1) \mid, \mid tset(P_2) \mid)} \quad (4)$$

$\mathsf{tsim}^{Simpson}(P_1,P_2) = \frac{1}{min(2,5)}$ = 0.5000
$\mathsf{tsim}^{Simpson}(P_1,P_3)$= 0.5000 $\mathsf{tsim}^{Simpson}(P_3,P_4)$= 0.7500

E. Difflib Similarity

Difflib similarity is defined as the matching words (M) multiplied by 2 and divided by the total number of words (T) in both sets [8]. We use multiset denoted by $tmset(\cdot)$ to find the similarity.
M= $\mid tmset(P_1) \cap tmset(P_2) \mid$
T = $\mid tmset(P_1) \mid + \mid tmset(P_2) \mid$
For the two concepts P_1 and P_2,

$$\mathsf{tsim}^{Difflib}(P_1, P_2) = 2 \times M/T. \quad (5)$$

$\mathsf{tsim}^{Difflib}(P_1,P_2)$= 2 × 1 / 7 =0.2857
$\mathsf{tsim}^{Difflib}(P_1,P_3)$= 0.3333 $\mathsf{tsim}^{Difflib}(P_3,P_4)$= 0.4000

F. Levenshtein Distance

Levenshtein Distance is based on the edit distance by counting the minimum number of operations (deletions, insertions and substitutions) required to transform the source string (L_1) into the target string (L_2) [9]. For the two concepts in Table I, we calculate $LevenshteinDistance(tlist(P_1),tlist(P_2))$ and the complete iteration is shown in Table III.

The distance is in the lower right hand corner of the matrix, i.e., distance=4. To get the similarity value from Levenshtein distance, we convert the distance into normalization denoted

TABLE III
LEVENSHTEIN DISTANCE BETWEEN TLIST(P_1), TLIST(P_2)

		tumor	dermis
	0	1	2
malignant	1	1	2
tumor	2	1	2
mesothelial	3	2	2
soft	4	3	3
tissue	5	4	4

by d_{norm} which is in the range of 0 and 1. *lendiff* = difference of length of the two lists.
$d_{norm}(L_1, L_2)$ = $\frac{d-lendiff}{min(|L_1|, |L_2|)}$

d_{norm}(tlist(P_1),tlist(P_2)) = $\frac{4-3}{2}$= 0.5000

After getting d_{norm}, we consider two ways to calculate the similarity:

$\mathsf{tsim}^{Levenshtein1}(P_1,P_2)$ = $1 - d_{norm}(tlist(P_1), tlist(P_2))$
$= 1 - 0.5000 = 0.5000$
$\mathsf{tsim}^{Levenshtein2}(P_1,P_2) = \left(\frac{1}{1+d_{norm}(tlist(P_1),tlist(P_2))}\right) \times$2-1
$= 0.3333$

$\mathsf{tsim}^{Levenshtein1}(P_1,P_3)$= 0.5000
$\mathsf{tsim}^{Levenshtein2}(P_1,P_3)$= 0.3333
$\mathsf{tsim}^{Levenshtein1}(P_3,P_4)$= 0.2500
$\mathsf{tsim}^{Levenshtein2}(P_3,P_4)$= 0.1429

IV. EXPERIMENTAL RESULTS ON SNOMED CT

SNOMED CT ontology denoted by $\mathcal{O}^{\text{SNOMED}}$ is written in Description Logic $\mathcal{EL}\mathcal{H}$ and it provides a standard terminology such that clinical and medical concepts are formally defined, systematically organized and are related to each other. Therefore, we choose the SNOMED CT as our experiments to generate possible doctors' judgments (preference profiles) for primitive concept and primitive role similarity. We use the DL version released in January 2005 which contains 364,461 concept names. In SNOMED CT, each concept name is uniquely identified by a concept ID (e.g. id=254735005), annotated with a short textual description (e.g. "Tumor dermis"), and equipped with a definition in description logic. The defined concepts are broadly categorized as subconcepts of one of 18 mutually exclusive top-level concepts. From the $\mathcal{O}^{\text{SNOMED}}$, we extract primitive concepts and roles for our work denoted by $\mathcal{O}^{pri}_{\text{SNOMED}}$.

For each top-level category C_i where $1 \leq$ i $\leq$ 18, we perform as the following.
1. Read primitive concept names and primitive role names from each category files as the lower case to reduce the case error and extracted the concept IDs, names and stored in array.
2. Make conventional text preprocessing tasks such as removing stop words because counting stop words in the intersection of sequences causes incorrect.

From SNOMED CT, we pick up only 50 primitive concepts samples for each category. We can totally generate 20825 number of pairs by considering only the distinct pairs (X,Y) from the same category C_i i.e. not include X = Y. In Table IV, the first column shows the top-level concept categories. The second and third column show the number of primitive sample concept names and pairs of concepts. Columns from four to ten are average and maximum values of primitive concept similarity and primitive role similarity generated by six similarity methods. We generate two kinds of primitive similarity degrees using two different ways of Levenshtein. The last two rows show the overall primitive similarity values based on all categories of each method. The last column shows the average of each category based on all methods. Among these methods, Jaccard gives smaller similarity values than others because it considers based on the union of two sets. Simpson gets the largest similarity values because it finds the overlap between sets.

From table IV, we can conclude about 73 % of the pairs are totally dissimilar (i.e. zero value for similarity) among 20825 pairs according to first five methods based on the average of concepts. Levenshtein distance gives 443 pairs of getting zero value more than other five methods because it consider ordering of lists. For example two primitive concepts "structure vesicular bursa sternohyoid muscle" and "pododerm structure" get 0.0000 from two cases of Levenshtein method while other methods yields 0.1667, 0.3162, 0.2857, 0.5 and 0.2857. Average execution time of each method for all pairs of primitive concepts(i.e. 20825) requires about 1.54 seconds that means each method can generate the primitive similarity values very fast. If we compute all concepts in SNOMED CT, it contains 364461 concepts so that it will take about 38 days for all total number of distinct pairs to generate primitive similarity.

A. Interesting Findings from the Experiments

According to above experiments, we get similarity degrees between pairs of primitive concepts and primitive roles using existing textual similarity methods. But we found one weak point from the experiments. Generating primitive similarity values using above six methods can not be true for some pairs of concepts because these methods treat the same weight for all positions of terms. Let consider the following three concepts P_5, P_6 and P_7.
P_5= Structure of capsule of kidney
P_6= Entire venulae rectae of medulla of kidney
P_7= Structure of capsule of red nucleus
If we look at pairs of (P_5,P_6) and (P_5,P_7), it is reasonable that similarity value of (P_5,P_6) $>$ (P_5,P_7) because (P_5,P_6) are parts of kidney and (P_5,P_7) are different parts of body structure. But existing six methods give unsuitable similarity values (P_5,P_6 $<$ P_5,P_7) as shown in Table V.

TABLE IV
EXPERIMENTAL RESULTS OF PRIMITIVE SIMILARITY DEGREES BASED ON DIFFERENT TEXTUAL METHODS ON SNOMED CT

SNOMED CT category C_i	Number of concepts	Number of Pairs	Primitive Similarity Measures (avg/ max)						
			tsim^{Jaccard}	tsim^{Cosine}	tsim^{Tversky}	tsim^{Simpson}	tsim^{Difflib}	tsim^{Levenshtein1} tsim^{Levenshtein2}	Average
Body structure	50	1225	0.103/0.714	0.163/0.833	0.162/0.833	0.188/0.833	0.161/0.833	0.150/0.833 0.100/0.714	0.146/1.000
Context-dependent	50	1225	0.031/0.667	0.050/0.817	0.049/0.857	0.059/1.000	0.048/0.800	0.056/1.000 0.037/1.000	0.047/1.000
Environment	50	1225	0.049/0.833	0.072/0.913	0.071/0.909	0.081/1.000	0.071/0.909	0.078/1.00 0.056/1.000	0.068/1.000
Event	50	1225	0.148/0.933	0.216/0.967	0.210/0.966	0.304/1.00	0.205/0.944	0.300/1.000 0.228/1.000	0.230/1.000
Finding	50	1225	0.026/0.750	0.039/0.866	0.039/0.923	0.047/1.000	0.038/0.857	0.038/1.000 0.028/1.000	0.036/1.000
Observable Entity	50	1225	0.060/0.833	0.095/0.913	0.094/0.909	0.120/1.000	0.093/0.909	0.105/1.000 0.071/1.000	0.091/1.000
Organism	50	1225	0.007/0.500	0.009/0.707	0.009/0.667	0.012/1.000	0.009/0.667	0.012/1.000 0.011/1.000	0.009/1.000
Physical Force	50	1225	0.119/0.800	0.177/0.894	0.179/0.889	0.195/1.000	0.176/0.889	0.193/1.000 0.137/1.000	0.168/1.000
Physical Object	50	1225	0.252/0.800	0.387/0.894	0.379/0.889	0.455/1.000	0.378/0.889	0.455/1.000 0.313/1.000	0.374/1.000
Procedure	50	1225	0.125/0.750	0.185/0.8666	0.184/0.857	0.201/1.000	0.184/0.857	0.199/1.000 0.141/1.000	0.174/1.000
Product	50	1225	0.013/0.667	0.020/0.817	0.020/0.800	0.023/1.000	0.020/0.800	0.023/1.000 0.016/1.000	0.019/1.000
Qualifier Value	50	1225	0.035/0.750	0.048/0.866	0.048/0.857	0.051/1.000	0.047/0.857	0.045/1.000 0.035/1.000	0.044/1.000
Social Concept	50	1225	0.028/0.800	0.040/0.894	0.039/0.889	0.048/1.000	0.038/0.889	0.051/1.000 0.042/1.000	0.041/1.000
Special Concept	50	1225	0.028/0.800	0.040/0.894	0.039/0.889	0.043/1.000	0.039/0.889	0.041/1.00 0.030/1.000	0.037/1.000
Specimen	50	1225	0.292/0.800	0.417/0.894	0.413/0.889	0.465/1.000	0.412/0.889	0.438/1.000 0.323/1.000	0.394/1.000
Staging Scale	50	1225	0.162/0.800	0.236/0.894	0.233/0.889	0.268/1.000	0.227/0.800	0.211/1.000 0.161/1.000	0.214/1.000
Substance	50	1225	0.002/0.600	0.003/0.750	0.003/0.750	0.003/0.750	0.003/0.750	0.003/0.750 0.002/0.600	0.003/1.000
Concept Average	850	20825	0.090/0.933	0.134/0.967	0.132/0.966	0.159/1.000	0.131/0.944	0.150/1.000 0.108/1.000	0.129/1.000
Roles	50	1225	0.019/0.500	0.029/0.707	0.028/0.667	0.035/1.000	0.028/0.667	0.032/1.000 0.024/1.000	0.028/1.000

TABLE V
SIMILARITY VALUES WITHOUT HEADWORD CONSIDERATION

Concepts	Jaccard	Cosine	Tversky	Simpson	Difflib	Leven1	Leven2
P_5, P_6	0.143	0.2582	0.25	0.33	0.25	0.33	0.2
P_5, P_7	0.4	0.5774	0.57	0.667	0.572	0.667	0.5

Therefore, we consider the important of headword of the noun phrase because all of the SNOMED CT concept names are noun phrase with prepositions. So, we add the important of headwords similarity denoted by $simHead$ and we use the basic similarity measure "Jaccard" for the similarity calculation of headwords.
$simHead(h_1,h_2) = \frac{h_1 \cap h_2}{h_1 \cup h_2}$.

According to headword idea of noun phrase, P_5 and P_6 have the same headword "kidney" and headword of P_7 is "red nucleus". Therefore, we modify each method by adding the multiplication of $simHead(h1, h2)$ to the similarity value of two concepts as the following.

$$simValueOf(P_1, P_2) \times simHead(h1, h2). \quad (6)$$

If the $simHead(h1, h2)$ gets zero, we multiply the simValueOf(P_1,P_2) by 0.1 in order to avoid getting zero values.
For Jaccard similarity,
simValueOf(P_5,P_6)$\times$ $simHead(h1, h2)$ = 0.143 $\times$ 1= 0.143
simValueOf(P_5,P_7)$\times$ $simHead(h1, h2)$ = 0.4 $\times$ 0.1= 0.04
It gives the more correct similarity degrees(P_5,P_6 $>$ P_5,P_7) as shown in Table VI.

TABLE VI
SIMILARITY VALUES WITH HEADWORD CONSIDERATION

Concepts	Jaccard	Cosine	Tversky	Simpson	Difflib	Leven1	Leven2
P_5, P_6	0.143	0.2582	0.25	0.33	0.25	0.33	0.2
P_5, P_7	0.04	0.057	0.057	0.067	0.057	0.067	0.05

Let consider others three concepts.
P_8= Skin structure of medial surface of fourth toe
P_9= Subcutaneous tissue structure of lateral surface of second toe
P_{10}= Entire flexor tendon and tendon sheath of fourth toe

Without headword consideration, P_8, P_9 gets more similarity values than P_8, P_{10} as shown in Table VII. But it should be $P_8, P_9 < P_8, P_{10}$ because P_8, P_9 are different parts of toe but both P_8, P_{10} are parts of fourth toe.

TABLE VII
SIMILARITY VALUES WITHOUT HEADWORD CONSIDERATION

Concepts	Jaccard	Cosine	Tversky	Simpson	Difflib	Leven1	Leven2
P_8, P_9	0.3	0.46	0.46	0.5	0.46	0.5	0.33
P_8, P_{10}	0.2	0.27	0.31	0.33	0.31	0.33	0.2

In the case of headword consideration, P_8 and P_{10} have the same headword "fourth toe" and P_9 have the headword "second toe".

For P_8,P_9, $simHead(h_1, h_2)$= simHead("fourth toe","second toe")= 1/ 3 =0.33
For P_8,P_{10}, $simHead(h_1,h_2)$= 1

For Jaccard similarity,
simValueOf(P_8,P_9)$\times$ $simHead(h1, h2)$ = 0.3 $\times$ 0.33= 0.1
simValueOf(P_8,P_{10})$\times$ $simHead(h1, h2)$ = 0.2 $\times$ 1= 0.2
Now, we get more correct similarity values P_8,P_9 $<$ P_8,P_{10} as shown in Table VIII.

TABLE VIII
SIMILARITY VALUES WITH HEADWORD CONSIDERATION

Concepts	Jaccard	Cosine	Tversky	Simpson	Difflib	Leven1	Leven2
P_8, P_9	0.1	0.15	0.15	0.17	0.15	0.17	0.11
P_8, P_{10}	0.2	0.27	0.31	0.33	0.31	0.33	0.2

Therefore, existing similarity methods by adding similarity of headwords can give more correct primitive measures than similarity methods without headword consideration.

V. CONCLUSION AND FUTURE WORK

This paper proposed to employ various textual similarity methods to get the similarity degrees of primitive concepts and primitive roles on SNOMED CT. We used different textual similarity methods to generate primitive similarity degrees because human experts also judge the similarity of primitive concepts based on equality of terms. So, obtained primitive similarity degrees can be regarded as possible human experts judgments (preference profile) on primitive concepts. From the experiments on SNOMED CT, we pointed out the important of headword consideration in order to get more correct similarity values for all pairs of SNOMED CT primitive concepts.

There are some directions for our future work. Firstly, we are going to apply deeply the important of headword by adding different weights to each terms for generating more correct similarity degrees for all pairs of primitive concepts on SNOMED CT. Secondly, we will combine new others similarity methods based on language model to generate different better primitive measures. Thirdly, we will cluster the primitive concepts of SNOMED CT based on similarity degrees from each method. Finally, we intend to perform human evaluation to measure the matching score between automatically-generated primitive similarity measures and medical experts' similarity judgments. We will prove primitive similarity measures using the modified different weights of headword consideration give the best similarity results among all implemented algorithms.

ACKNOWLEDGMENT

This research is partially supported by Thammasat University Research Fund under the TU Research Scholar, Contract No. TOR POR 1/13/2558. The first author is supported by Sirindhorn International Institute of Technology under an Excellent Foreign Student (EFS) scholarship.

REFERENCES

[1] T. Racharak, B. Suntisrivaraporn and S. Tojo. sim^{π}: A Concept Similarity Measure under an Agents Preferences in Description Logic $\mathcal{ELH}$. In Proceedings of the 8th the International Conference on Agents and Artificial Intelligence (ICAART 2016). Rome, Italy, 2016.

[2] S. Tongphu and B. Suntisrivaraporn: Algorithms for Measuring Similarity Between $\mathcal{ELH}$ Concept Descriptions: A Case Study on SNOMED CT. Journal of Computing and Informatics, Vol-20, Jul-8, 2015.

[3] B. Suntisrivaraporn: A Similarity Measure for the Description Logic EL with Unfoldable Terminologies. In: INCos, pages 408-413, 2013.

[4] S. Niwattanakul, J. Singthongchai, E. Naenudorn, S. Wanapu: Using of Jaccard Coefficient for Keywords Similarity. In: Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS), Vol-1, March, Hong Kong 2013.

[5] K.P.N.V.Satya sree, Dr.J V R Murthy: Clustering Based on Cosine Similarity Measure. In: International Journal of Engineering Science and Advanced Technology (IJESAT),Vol-2, Issue-3, pp.508- 512, 2012.

[6] S. Jimenez, C. Becerra, A. Gelbukh, Softcardinality-core: Improving Text Overlap with Distributional Measures for Semantic Textual Similarity. In: Second Joint Conference on Lexical and Computational Semantics (*SEM), Vol-1, pp.194-201, June, Atlanta, Georgia, 2013.

[7] J. Choi, T. Oh, and I. So Kweon, Human Attention Estimation for Natural Images: An Automatic Gaze Refinement Approach. Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea, 12 Jan, 2016.

[8] K. Wolk, K. Marasek: A Sentence Meaning Based Alignment Method for Parallel Text Corpora Preparation. In: Rocha, A., Correia, A.M., Tan, F.B., Stroetmann, K.A. (eds.)New Perspectives in Information Systems and Technologies, vol 1, pp. 229-237, Springer, Switzerland 2014.

[9] A. McCallum: String Edit Distance (and intro to dynamic programming) Computational Linguistics, Spring 2006.

[10] Wael H. Gomaa and Aly A. Fahmy: A Survey of Text Similarity Approaches: International Journal of Computer Applications, Vol-68, No.13, April 2013.