Siamese KG-LSTM: A deep learning model for enriching UMLS Metathesaurus synonymy

Tien T. T. Tran[1], Sy V. Nghiem[1], Van T. Le[1], Tho T. Quan[1]
Vinh Nguyen[2], Hong Yung Yip[3], Olivier Bodenreider[2]
[1]Computer Science and Engineering, Ho Chi Minh University of Technology, HCMC, Vietnam
[2]National Library of Medicine, National Institute of Health, Bethesda, MD, USA
[3]Artificial Intelligence Institute, University of South Carolina, Columbia, SC, USA
Email: [1] {tien.tran1727039, sy.nghiemcongnghe17, ltvan, qttho}@hcmut.edu.vn
[2] vinh.nguyen@nih.gov, obodenreider@mail.nih.gov,
[3] hyip@email.sc.edu

Abstract—The Unified Medical Language System, or UMLS, is a repository of medical terminology developed by the U.S. National Library of Medicine for improving the computer system's ability of understanding the biomedical and health languages. The UMLS Metathesaurus is one of the three UMLS knowledge sources, containing the medical terms and their relationships. Due to the rapid increase in the number of medical terms recently, the current construction of UMLS Metathesaurus, which heavily depends on lexical tools and human editors, is error-prone and time-consuming. This paper takes advantage of the emerging deep learning models to learn the similarities between the pairs of biomedical terms archived in UMLS for identifying synonymous in the Metathesaurus. Our learning approach focuses a target subset of specific terms instead of the whole Metathesaurus corpus. Particularly, we train the models with biomedical terms from the Disorders semantic group. To improve the effectiveness of the model, we enrich the input with different enrichment strategies, including synonyms and hierarchical relationships from source vocabularies. Our deep learning model adopts the Siamese KG-LSTM (Siamese Knowledge Graph - Long Short-Term Memory). The experimental results show that our approach yields excellent performance when handling the task of synonym detection for Disorders semantic group in the Metathesaurus. This illustrates the potential of applying machine learning model with synonyms in source enriching strategy into the current UMLS Metathesaurus construction process. Although the work in this paper focuses only on specific semantic group of Disorders, we believe that the proposed method can be applied to other semantic groups in the UMLS Metathesaurus.

Index Terms—Unified Medical Language System, UMLS Metathesaurus, Semantic Text Similarity, Siamese KG-LSTM, Deep Learning, Knowledge Graph

I. INTRODUCTION

The *Unified Medical Language System* (UMLS) is a rich unified medical vocabulary repository, created by the U.S. National Library of Medicine to enable interoperability between computer systems [1]. For example, the same concept of *Addison's Diseases* can be defined by the term *Primary hypoadrenalism* in the category of *Medical Dictionary for Regulatory Activities* (MedDRA) or *Primary adrenocortical insufficiency* in *the 10th revision of the International Statistical Classification of Diseases* (ICD-10) [2]. The UMLS unifies medical languages, thereby supporting the automated process of integrating clinical information from different health systems.

There are three knowledge sources in UMLS, including *Metathesaurus*, *Lexical Resources*, and *Semantic Network*. The UMLS Metathesaurus is a large, multi-purpose, and multi-lingual database which contains the biomedical terms as well as their definitions, attributes, hierarchies and relationships. Thanks to continuous medical advances, the Metathesaurus database adds thousands of new concepts, names, relationships each year. The numbers of AUI (unique atomic identifier) and CUI (unique conceptual identifier) in Metathesaurus database from 2015 to 2019 are depicted in Fig. 1.
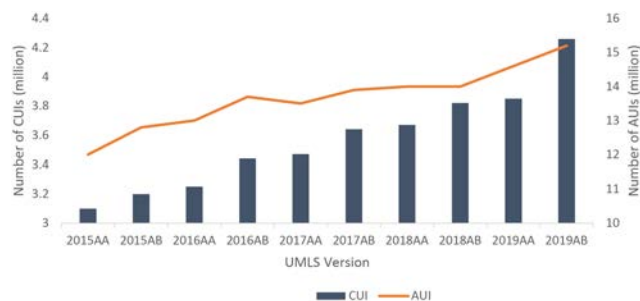

Fig. 1. Number of Metathesaurus AUI and CUI from 2015 to 2019.

The construction of the Metathesaurus consists of several processes. The first step is *Source inversion*, which needs to be carried out by the UMLS editors [3]. One of highly intellectual tasks in this process is that identifying synonymous and non-synonymous terms from UMLS Metathesaurus terminologies to select suitable unique conceptual identifier (CUI) for each term in the new vocabulary to be inserted to the UMLS. Along with the development of the Metathesaurus database (Fig. 1), with more than 15 million AUIs and 4.2 million CUIs in version 2019AB, the current manual process is error-prone and time-consuming.

Thanks to the recent developments of deep learning models in understanding and manipulating data, especially in the medical and healthcare domain [4]. In this paper, we propose a method using deep learning models to learn the similarity in the terminologies. First, the pair of input terms to be consid-

ered are *embedded* by Biowordvec and then enter the Siamese model with LSTM network to create the corresponding *lexical vectors*. Next, to enrich the context of the vocabulary, each *lexical vector* can be concatenated with a *context vector*, generated by using the *node2vec* technique to embed the graph of local relationships between the input term and related terms in the dataset. The similarity of the two combined vectors, after fed to a 2-layer dense feed-forward network, is measured by a Manhattan distance. Based on the measured distance, one can infer the similarity of the corresponding term pair in a straightforward manner.

Since the training of such deep leaning model is computationally expensive, we develop a strategy to avoid training on the whole UMLS Metathesaurus corpus. Particularly, we train the models with terms belonging to the semantic group *Disorders* which accounts for the second highest proportion in Metathesaurus. Training on such a target subset not only reduces time for generating features but also allows a wide range of options for computing resources.

Our contribution is that we applied a novel method which enhances traditional deep learning model Siamese MaLSTM with knowledge features extracted from contextual knowledge graphs to build the Siamese KG-LSTM model for assessing the semantic similarity between two biomedical terms. In the application of the UMLS Metathesaurus, the model could identify synonymy and non-synonymy between UMLS terms, thereby giving suggestions about which concept a new term can be assigned to.

The rest of the paper is organized as follows. In Section II, a brief definition of semantic text similarity and related works in this fields are described. In Section III, we propose a framework for building the similarity measuring model. In Section IV, the experiment and evaluations from real data are presented. In Section V, we discuss about the results. Finally, Section VI provides our conclusion.

II. RELATED WORK

The task of measuring the semantic similarity between words and sentences, also known as *Semantic Text Similarity*, is a field that has received a great deal of attention and widely studied in Natural Language Processing (NLP) due to its important role in various tasks such as semantic search, text summary, answer questions, grouping of documents and identify synonyms.

There are three main approaches to measure similarity between words: string based, knowledge based and corpus based (Fig. 2) [5]. Almost all recent researches focus on the latter because it brings more strong points and achieve better results in measuring the semantic similarity. For tackling our problem, we also propose the solution based on the corpus-based approach combined with knowledge based to enhance the model accuracy.

- **String based**: Similarity is measured by comparing two character strings. In the research of Esko Ukkonen et al, they use the distance q-gram to estimate the similarity of
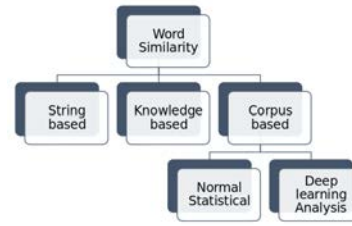

Fig. 2. Methods of measuring word similarity.

two chains based on the number of "common substring length q" [6].

- **Knowledge based**: Finding similarity by building a semantic network that expresses the content and relationships of words. Hongzhe Liu and Pengfei Wang used WordNet, a lexical network of vocabulary, to determine synonyms between very short texts or sentences without using any external corpus [7].

- **Corpus based**: Analyzing the coexistence between words in a large corpus to accurately evaluate similarity between words. There are two ways to analyze a corpus using normal statistical analysis and using deep learning model to find semantic representation of the word. In 1998, Thomas K Landauer et al. proposed the Latent semantic analysis (LSA) method, in which each word is represented by a vector based on statistical calculations. To build these vectors, a large text is analyzed and a word matrix is built. In this matrix, words and paragraphs are represented in corresponding rows and columns. Moreover, an algorithm called Singular Value Decomposition (SVD) is applied to reduce the number of dimensions. Based on the constructed vocabulary vector, the similarity of words is calculated by cosine similarity [8]. In another research, Tomas Mikolov et al. used deep learning to train a model to predict a word with a given surrounding word (CBOW Continuous Bag of Word model) or context words from a given word (Skipgram). This model generates a representing vector depends on the appearance of the word in the corpus. Cosine similarity among the vectors was used to measure word similarity [9]. Based on the deep LSTM network, Mueller et al. proposed the MaLSTM model to learn similar sentences. The input is a sentence represented by words' vectors which have the same size. Moreover, a label representing the similarity is assigned to each pair of sentences. The model uses LSTM to read the input vectors and uses the last hidden layer as the vector representing the sentence. The new genereted representative vector is used to learn semantic similarity [10].

Moreover, some projects show that the combination of corpus based and knowledge based help to improve in identifying similarity of sentences instead of using only single methods. For example, Nitish Aggarwal et al. compares the combination of corpus based ESA and knowledge based WordNet with single use of WordNet on data sets [11]. For semantic similarity

analysis, the research group of Hong Yung Yip uses Siamese MaLSTM combines with vocabulary enrichment by the Bag of Word method on the UMLS Metathesaurus dataset [2].

Therefore, in this article, we use corpus based (based on the Siamese LSTM network model [10]) in combination with knowledge based (vocabulary enriched by building local relationship graphs [12] and using node2vec embedding [13] to create context vectors) to build Siamese KG-LSTM model, which allows deeply analyze for predicting similar terms in the Metathesaurus dataset.

III. OUR PROPOSAL FRAMEWORK

In this article, we construct a Siamese KG-LSTM model to learn the similarity between biomedical term pairs in the Metathesaurus. Similar to [2], the input terms are extracted from the MRCONSO file to generate synonymous and non-synonymous pairs. Different strategies includes adding source synonyms and hierarchical context in addition to the lexical term itself. Manhattan distance is used to measure the similarity, thereby identifying which medical terms are synonymous. While [2] used the UMLS release 2019AA for the evaluation, we used the UMLS release 2019AB.

A. Datasets

The study was conducted with AUIs which language is English and semantic group is *Disorders* (Fig. 3) of all lexical sources extracted in the Metathesaurus version 2019AB, released on November 4th, 2019.
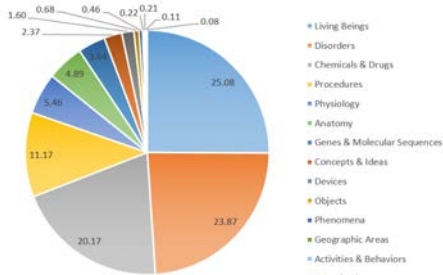

Fig. 3. Percentage of AUI according to the semantic group of Metathesaurus 2019AB.

The final dataset comprises **2,580,001** AUIs and **660,170** CUIs.

B. Feature engineering

Generating synonymous pairs. In the Metathesaurus, AUIs assigned to the same CUI are synonymous. Therefore, every two AUIs in the same CUI will produce a synonymous pair. For example, the synonymous terms *Addison disease*, *Primary hypoadrenalism*, *Primary adrenocortical insufficiency*, and *Addison's disease (disorder)* are assigned to the same concept C0001403. There are six possible synonymous pairs generated:

- *Addison disease* and *Primary hypoadrenalism*.
- *Addison disease* and *Primary adrenocortical insufficiency*.
- *Addison disease* and *Addison's disease (disorder)*.

- *Primary hypoadrenalism* and *Primary adrenocortical insufficiency*.
- *Primary hypoadrenalism* and *Addison's disease (disorder)*.
- *Primary adrenocortical insufficiency* and *Addison's disease (disorder)*.

From the generated dataset, we create a CUI-AUI dictionary for every CUI and its assigned AUIs. For every CUI in the dicCUI_AUI, any AUI pair from its assigned AUIs is synonymous. As a result, there are **18,939,003** synonymous pairs created.

Generating non-synonymous pairs. In the Metathesaurus, AUIs from different CUIs are not synonymous. However, because the number of non-synonymous pairs is too large, it is computationally infeasible to generate all non-synonymous pairs. Jaccard Index, as showed in Equation (1), can be used to limit the number of non-synonymous pairs. Only AUI pairs that have a Jaccard Index greater than a certain threshold are included in the set of non-synonymous pairs [2]. For example, we consider A30242010 *Product containing para-aminobenzoic acid* and A29490109 *Product containing sulfuric acid*. There are three similar words between two strings: *Product*, *containing* and *acid*. The total number of the words in the strings is five, including *Product*, *containing*, *acid*, *para-aminobenzoic* and *sulfuric*. Jaccard Index of these two AUIs equals to the number of similar words divide total number of words, which is 3/5 = 0.6.

$$Jaccard(A, B) = \frac{|A \cap B|}{|A \cup B|} \qquad (1)$$

From the generated dataset, for each AUI, its non-synonymous list is created by selecting other AUIs, which have string length in range of equal to triple its length, not belongs to its CUI (eliminating synonymous pairs) and the Jaccard indexes with it are greater than 0 (eliminating pairs which do not have any lexical similarity). As a result, there are **125,918,700** non-synonymous pairs created.

Generating input terminologies: The non-synonymous pairs are sorted from largest to smallest Jaccard index, then the pairs with the highest Jaccard indexes will be selected to combine with synonymous pairs to generate input datasets. The Jaccard cut-off is selected based on the ratio of the total number of synonymous and non-synonymous pairs. If the ratio is 1:1, the Jaccard cut-off will be 0.33. If the ratio is 1:3, the Jaccard cut-off will be 0.2.

The number of synonymous and non-synonymous pairs sampled in the 1:1 and 1:3 ratio with Jaccard Index cut-off is presented in Table I.

TABLE I
SUMMARY OF THE NUMBER OF SYNONYMOUS AND NON-SYNONYMS PAIRS

Ratio	Synonym	Non-Synonym	Jaccard cut-off
1:1	18,939,003	18,939,003	0.33
1:3	18,939,003	56,817,009	0.2

C. Enriching Strategies

In the Metathesaurus, if a term puts in different contexts, it can reflect different meanings. For example, the term *nail* means *a part of the human body* and *a device*. Therefore, depending merely on the lexical term to predict synonyms may be not enough, and [2] also showed that it is necessary to add the relevant contexts to clarify the meaning of the term.

The Metathesaurus comprises of many relationships between different concepts. Most of these relationships comes from individual vocabulary source, some were added during the construction of the Metathesaurus, some were contributed by users. The relationships are extracted from the MRREL of Metathesaurus.

Considering the current manual process, when a new term is added, UMLS editors will evaluate the term based on relationships in its source. To simulate this process, besides experiment 1 which includes only lexical term, we designed additional experiments 2, 3 and 4 with enriching context (Fig. 4).
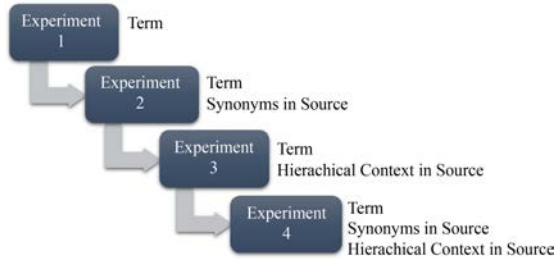

Fig. 4. Contextual enrichment experiment model design.

- **Experiment 1**: only lexical characteristics of input terms.
- **Experiment 2**: The input terms are contextualized with their synonyms in its vocabulary source extracted by the SY relationships in the MRREL file.
- **Experiment 3**: The input terms are contextualized with their parents, children, siblings in its vocabulary source extracted by the PAR, CHD, SIB relationships.
- **Experiment 4**: is the combination of Experiment 2 and Experiment 3.

Firstly, from the generated dataset, we create *relation dictionaries* including dicAUI_PAR, dicAUI_CHD, dicAUI_SIB with AUIs as keys and the sets of their parents, children, and sibling relationships as values, respectively. Secondly, the networkx [12] is adopted to construct a graph consisting local relationships for each AUI. The central node of the graph is the AUI that needs to contextualize and other nodes are AUIs extracted from the *relation dictionary* depending on the vocabulary enrichment strategy. After that, the node2vec [13] is used to generate the 200-dimensional embedding vector for the central AUI in the graph. The central AUI with its corresponding vector are saved to *embedding dictionaries* such as dicEMB_SYM, dicEMB_HIER and dicEMB_ALL based on different vocabulary enrichments.

D. Semantic Similarity Measuring Model

The deep learning model proposed to solve our task is based on the Siamese LSTM structure. Unlike the traditional neuron networks that receive only one input at a time, Siamese network receives a pair of input for learning. The neural network used in this model is the long short term memory network (LSTM), a special type of recurrent neural network (RNN) that is significant for its ability to learn longer string data and mitigating the vanishing gradient problem.

The noises such as special characters, punctuation, spaces, and carriage returns in the input pairs needs to be removed before feeding them into the model. After that, each word in the term of the pair will be converted into token by looking up a dictionary, which key is every single word in the research dataset and value is the token representing for that word. As a result, for each input term, a vector with a dimension equal to the number of words in the string is created. Based on the word length distribution scheme in the dataset, most of the terms's length are less than or equal to 30 (Fig. 5). Therefore, we apply padding with zeroes in the beginning or truncating at the end of the input to ensure every terminology has the same 30-dimensional representative vector.
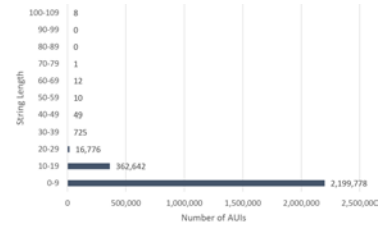

Fig. 5. Length distribution in final dataset.

The input vectors are fed into the word embedding layer to create the matrix representing the sequence of terms. The initial weights, generated from the pre-trained BiwordVec. BioWordVec [14], is a set of biomedical word vectors generated by using subword methods on PubMed biomedical vocabulary dataset and MeSH (medical subject headings) data. The subword embedding model enables BioWordVec to learn rare vocabulary words, such as *deltaproteobacteria*. Therefore, BioWordVec could better improve the quality of word embedding for medical terms.

The output matrix of the word embedding, which has 30 rows and 200 columns, is put into the LSTM layer with 50 units to learn the semantic characteristics of the string based on the word orders through time.

- **Baseline model**: The pair of *lexical vectors*, which are outputs of LSTM layer, is measured by using the Manhattan distance function. If the value is greater than or equal to 0.5, the inputs are synonymous; otherwise, they are non-synonymous (Fig. 6).
- **Siamese KG-LSTM model**: The *lexical vector*, which is output of LSTM layer, will be concatenated with its corresponding *context vector* extracted from the *embedding dictionary* based on different vocabulary enrichment strategies. The concatenated vector then enters the two

dense layers to create a *combined vector*. The model's output is the similarity score between the two *combined vectors*, which is measured by Manhattan distance function. If the value is greater than or equal to 0.5, the inputs are synonymous; otherwise, they are non-synonymous (Fig. 7).
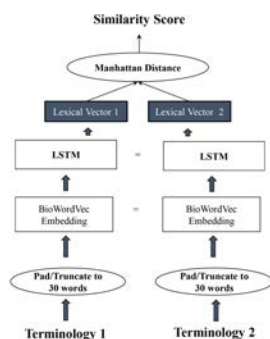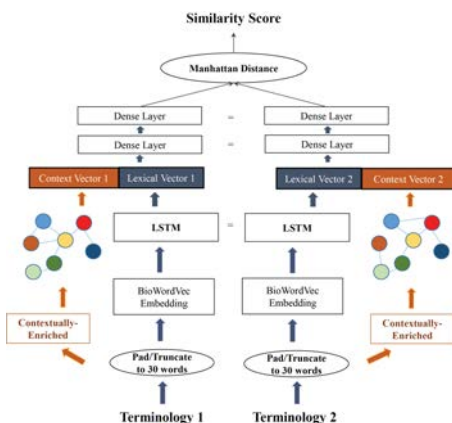

Fig. 6. Baseline model design.

Fig. 7. Siamese KG-LSTM model design.

Table II summarizes the parameters used for Baseline model (Exp.1) and Siamese KG-LSTM model (Exp.2,3,4).

TABLE II
MODEL PARAMETERS

Parameter	Baseline	Siamese KG-LSTM
	Exp.1	Exp.2,3,4
Framework	Keras 2.2.4 with Tensorflow backend 2.1.0	
Word Vector Size	200	
Maximum Input Length	30	
Embedding	BioWordVec Trainable	BioWordVec + Node2Vec Trainable
LSTM Hidden Units	50	
LSTM Activation	Tanh	
Dense Layer 1	-	128 units with ReLU activation
Dense Layer 2	-	50 units with ReLU activation
Weights and Biases	Xavier Initialization	
Optimizer	Adam	
Learning Rate	0.001	
Loss Function	Mean Squared Error (MSE)	
Batch Size	256	
Epochs	9	
Train/Validation Ratio	80/20	

Each experiment is trained with 1:1 and 1:3 ratios between the number of synonymous and non-synonymous pairs. We use a GPU server with 4 Nvidia Tesla V100-SXM2 and a CPU server with 32 cores from Intel(R) Xeon(R) CPU E5-4640 for the experiments.

IV. EXPERIMENT AND EVALUATION

The performances of the semantic similarity measuring models are evaluated based on classifying synonyms and non-synonyms when adding a new term into the dataset. The supervised learning metrics include validation accuracy, precision, recall, F1-Score, specificity, and false-positive rate.

The experimental results on testing dataset with the ratio 1:1 between the number of synonymous and non-synonymous pairs are shown in Table II.

TABLE III
EXPERIMENTAL RESULTS ON TESTING DATASET WITH RATIO 1:1

Model Metrics	Exp.1 Term	Exp.2 Term + SY[a]	Exp.3 Term + HC[b]	Exp.4 Term + SY + HC
Accuracy	0.9712	**0.9760**	0.9737	0.9740
Precision	0.9746	**0.9767**	0.9749	0.9751
Recall	0.9676	**0.9753**	0.9725	0.9729
F1 score	0.9711	**0.9760**	0.9737	0.9740
Specificity	0.9748	**0.9767**	0.9750	0.9752
False Positive Rate	0.0252	**0.0233**	0.0250	0.0248

[a]SY: Synonyms in Source, [b]HC: Hierachical Context in Source.

As shown in Table III, the experiment with only terms (Exp.1) can identify the synonymous pairs with 97.12% accuracy. Adding hierarchical relationships in source vocabularies (Exp.3) does not seem to improve the efficiency, since the accuracy of this strategy is approximate to the Exp.1, at 97.37%. The combination of source synonyms and hierarchical context in Exp.4 has 97.40% accuracy, slightly higher than the first experiment. Meanwhile, enriching with solely synonyms in source relationships (Exp.2) is likely to be the best model with highest performance results, in which the accuracy is 97.60%.

TABLE IV
EXPERIMENTAL RESULTS ON TESTING DATASET RATIO 1:3

Model Metrics	Exp.1 Term	Exp.2 Term + SY[a]	Exp.3 Term + HC[b]	Exp.4 Term + SY + HC
Accuracy	0.9794	**0.9823**	0.9780	0.9802
Precision	0.9778	**0.9786**	0.9734	0.9758
Recall	0.9387	**0.9501**	0.9377	0.9442
F1 score	0.9579	**0.9641**	0.9552	0.9597
Specificity	0.9929	**0.9931**	0.9915	0.9922
False Positive Rate	0.0071	**0.0069**	0.0085	0.0078

[a]SY: Synonyms in Source, [b]HC: Hierachical Context in Source.

Compare to the results in Table III, the results of 1:3 ratio in Table IV is higher than those of 1:1 ratio. Besides, the pattern of enrichment strategies between two ratios is the same, which the highest efficient strategy is adding synonyms from source enrichment (Exp.2), with 98.23% accuracy.

V. Discussion

Table V shows the insights about the efficiency in classifying synonymous and non-synonymous pairs of Siamese KG-LSTM model from Experiment 2 ratio 1:3.

True Positives	
apnoea	not breathing
Malignant neoplasm	Cancer
Sulfite oxidase deficiency syndrome	Sulphocysteinuria
True Negatives	
Adverse reaction to sulfaurea	Adverse reaction to tetracaine
wound of back of head with bleeding tissue (physical finding)	wound of right anterior chest with venous bleeding (physical finding)
puncture of left lower leg with swelling (physical finding)	puncture of right upper back by needle (physical finding)
False Positives	
Low pressure	Decreased pressure
Myelofibrosis	Primary Myelofibrosis
Pharynx--Tumors	Pharynx--Cancer
False Negatives	
anomaly of chromosome pair ring chromosome 19 syndrome	(19)r syndrome
L.eye-SVI, B.eye-unspecified	Lesser eye: severe visual impairment, Better eye: unspecified (disorder)
Acute Hepatitis A	Hepatitis infectious

From Table V, when observing intuitively some specific cases, the model can predict correctly the pairs *apnoea* and *not breathing* to be synonymous although they do not share any similar strings.

Additionally, the model can identify the term *Adverse reaction to sulfaurea* not synonymous with *Adverse reaction to tetracaine* although they have four similar strings.

However, there are some mistakes in identifying terms having the similar strings in different CUIs to be synonymous while the actual class of pairs is not. For example, term *Myelofibrosis* which belongs to both CUI *C0026987* and *C0001815* in Metathesaurus dataset. Therefore, *Myelofibrosis* in CUI *C0026987* is not synonymous with AUIs in CUI *C0001815*, such as *Primary Myelofibrosis*, while *Myelofibrosis* in CUI *C0001815* is synonymous with *Primary Myelofibrosis*. This leads to the misunderstanding of the model.

Similarly, the model is more likely to classify non-synonymous for acronym term and the full terms, such as *L.eye-SVI, B.eye-unspecified* and *Lesser eye: severe visual impairment, Better eye: unspecified (disorder)* because of their less common strings, although they are actual synonymous.

VI. Conclusion

The paper proposes a new model architecture based on the traditional Siamese by adding semantic features extracted from the knowledge graph, named Siamese KG-LSTM. Applying dataset extracted from Metathesaurus 2019AB, which have *English* language and specific semantic group *Disorders*,

Siamese KG-LSTM with synonyms in source enrichment strategy can slightly improve the identification of semantic similarity among terms.

In addition, our study results show the potential of applying Siamese KG-LSTM model with the strategy of enriching the lexical terms with synonyms and hierarchies from source vocabularies in the assessment of semantic similarity between biomedical terms.

Although this article only focuses on specific semantic group *Disorders*, we believe that the proposal method is promising and can be applied to other semantic groups in UMLS Metathesaurus.

References

[1] O. Bodenreider, "The Unified Medical Language System (UMLS): integrating biomedical terminology", Nucleic Acids Research, vol. 32, no. 90001, pp. 267D-270, 2004.

[2] H.Y. Yip, V. Nguyen and O. Bodenreider, "Construction of UMLS Metathesaurus with Knowledge-Infused Deep Learning", in Proceedings of the Second International Workshop on Contextualized Knowledge Graphs (CKG 2019), 2019.

[3] S. Liu, "Enabling Electronic Healthcare Information Exchange", IT Professional, vol. 9, no. 6, pp. 17-23, 2007.

[4] A. Esteva et al., "A guide to deep learning in healthcare", Nature Medicine, vol. 25, no. 1, pp. 24-29, 2019.

[5] M. Farouk, "Measuring Sentences Similarity: A Survey", Indian Journal of Science and Technology, vol. 12, no. 25, pp. 1-11, 2019.

[6] E. Ukkonen, "Approximate string-matching with q-grams and maximal matches", Theoretical Computer Science, vol. 92, no. 1, pp. 191-211, 1992.

[7] H. Liu and P. Wang, "Assessing Sentence Similarity Using WordNet based Word Similarity", Journal of Software, vol. 8, no. 6, 2013.

[8] T. Landauer, P. Foltz and D. Laham, "An introduction to latent semantic analysis", Discourse Processes, vol. 25, no. 2-3, pp. 259-284, 1998.

[9] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space", in Proc. of ICLR Workshop, 2013.

[10] J. Mueller and A. Thyagarajan, "Siamese recurrent architectures for learning sentence similarity," in Proc. 30th AAAI Conf. Artif. Intell., 2016, pp. 2786–2792.

[11] N. Aggarwal, K. Asooja and P. Buitelaar, "DERI&UPM: Pushing corpus based relatedness to similarity: Shared task system description", 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, 2012.

[12] Aric A. Hagberg, Daniel A. Schult and Pieter J. Swart, "Exploring network structure, dynamics, and function using NetworkX", in Proceedings of the 7th Python in Science Conference (SciPy2008), 2008.

[13] A. Grover and J. Leskovec, "node2vec", in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.

[14] Y. Zhang, Q. Chen, Z. Yang, H. Lin and Z. Lu, "BioWordVec, improving biomedical word embeddings with subword information and MeSH", Scientific Data, vol. 6, no. 1, 2019.