Enhance Accuracy of Hierarchical Text Categorization Based on Deep Learning Network Using Embedding Strategies

Chanantip Saetia, Peerapon Vateekul
Chulalongkorn University Big Data Analytics and IoT Center (CUBIC), Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University
Bangkok, Thailand, 10330
chanatip.s@student.chula.ac.th, peerapon.v@chula.ac.th

***Abstract*—Hierarchical text categorization is a task that aims to assign predefined categories to text documents with hierarchical constraint. Recently, deep learning techniques has shown many success results in various fields, especially, in text categorization. In our previous work called Shared Hidden Layer Neural Network (SHL-NN), it has shown that sharing information between levels can improve a performance of the model. However, this work is based on a sequence of unsupervised word embedding vectors, so the performance should be limited. In this paper, we propose a supervised document embedding specifically designed for hierarchical text categorization based on Autoencoder, which is trained from both words and labels. To enhance the embedding vectors, the document embedding strategies are invented to utilize the class hierarchy information in the training process. To transfer the prediction result from the parent classes, the shared information technique has been improved to be more flexible and efficient. The experiment was conducted on three standard benchmarks: WIPO-C, WIPO-D and Wiki comparing to two baselines: SHL-NN and a top-down based SVM framework with TF-IDF inputs called "HR-SVM." The results show that the proposed model outperforms all baselines in terms of F1 macro.**

Keywords—Hierarchical Multi-Label Classification; Deep Learning; Text Categorization;

I. INTRODUCTION

Hierarchical Text Categorization (HTC) is a text categorization, which each document can be assigned to many labels or categories and the labels are organized in a hierarchy. Higher level classes refer to be more general classes and lower level classes refer to be more specific classes. HTC has been becoming popular, so there are many prior attempts that tried to apply traditional classification techniques on a sparse document representation like word count and TF-IDF. However, this kind of representation ignores the word sequence, so the prediction performance should be limited.

Recently, deep learning has shown promising results in various fields, such as speech recognition, natural language processing, text categorization. The word and document embedding technique is one the most success deep learning xalgorithms. It aims to create a dense representation of word and documents by taking a word sequence into account. Since the output vector is dense, it can avoid the curse of dimensionality that occurs in sparse representations. The popular one of the word embedding is well-known as Word2Vec [1].

In the text categorization task, Y. Kim [2] applied a convolutional neural network (CNN) on a sequence of word embedding vectors. In the hierarchical classification domain, since there is a hierarchical relationship between parent and child classes, R. Cerri et al [3] proposed to induce a multi-layer perceptron network per each hierarchical class level called "Hierarchical Multi-Label Classification using Local Multi-Layer Perceptron (HMC-LMLP)." It tried to transfer information from a parent class to its subclasses. In our previous work [4], we proposed to induce a deep learning network specifically for HTC called "Shared Hidden Layer Neural Network (SHL-NN)." It induces one network of CNN per each hierarchical level along with our sharing information strategy. However, the embedding vectors in SHL-NN is based on Word2Vec, which is an unsupervised learning, and they are not designed for hierarchical problem. So, it is possible to further improve a prediction performance of SHL-NN.

In this work, we aim to improve the classifier for HTC. First, we build a new architecture on HMC-LMLP in order to provide more efficient sharing information. Second, the supervised document embedding vector is chosen rather than a sequence of unsupervised word embedding vectors as an input to the model. In the document embedding process, we also propose several novel training strategies based on a hierarchal constraint and then select the best one. Finally, our proposed method is compared to two baselines: (*i*) SHL-NN whose features is a sequence of word embedding with CNN and (*ii*) a TF-IDF based model called HR-SVM [5] that based on LIBSVM [6].

The paper is organized as follows. Section II is background knowledge including hierarchical classification, supervised document embedding, and SHL-NN. The details of the proposed method are described in Section III. The experiments setup and preprocessing datasets are shown in Section IV. Then, the results are discussed in Section V. Finally, the paper is concluded in Section VI.

978–1–5386–5538–2/18/$31.00 ©2018 IEEE

II. Background Knowledge

A. Hierarchical Classification

Hierarchical classification is a classification task where classes are organized in a structure like Tree or Directed Acyclic Graph (DAG). The hierarchical relationships contain the transitive relation and the asymmetric relation but do not include the reflexive relation. With a tree structure, only one class can be assigned to be a parent class of each classes. Let C is a finite set of all classes. We define all relationships below.

1) *Irreflexive:* $\forall c_i \in C,\ c_i \prec c_i$

2) *Assymetric:* $\forall c_i, c_j \in C,\ c_i \prec c_j \rightarrow c_j \prec c_i$

3) *Transitive:* $\forall c_i, c_j, c_k \in C,\ c_i \prec c_j, c_j \prec c_k \rightarrow c_j \prec c_k$

Be aware that, any data that belongs to a subclass will also belong to all ancestor classes of that subclass. Therefore, this problem is typically a multi-label classification.

From a previous research in an existing hierarchical classification [7], a local classification approach which is the composition of classifiers predicting some part of a hierarchy was proposed. Since classifiers are combined with hierarchical constraint, the subclass could not be predicted if any super classes were negative. This approach preserves the natural constraint of hierarchical classes. It is compatible with any classifier, but it may lead to the blocking problem.

B. Supervised Document Embedding (DocTag2Vec)

In machine learning, an input of most algorithms needs to be a fixed size vector. In NLP domain, many previous works need to create a fixed sized representation of each raw document to as an input for the model. There are two kinds of representations: sparse and dense. TF-IDF and word count are two most common sparse representations; however, they always face a problem of the curse of dimensionality because there can be hundreds of words in a corpus. Currently, Q. Le and T. Mikolov present a new dense representation called "Doc2Vec" [8], which solves the curse of dimensionality problem occurred in a sparse representation.

To accomplish a document tagging task, S. Chen et al [9] proposed a supervised method of Doc2Vec called "DocTag2Vec." Tags of each document is employed as an input of a model in a training process instead of using only words from each document. The architecture of DocTag2Vec is shown in Fig. 1.

For HTC, we aim to improve DocTag2Vec by utilizing a hierarchical constraint to construct an enhanced document embedding vector.

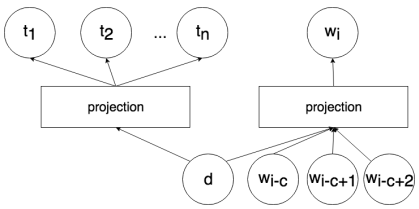

Fig. 1. The model architecture of DocTag2Vec

C. Hierarchical Multi-label Classification Using Local Multi-Layer Perceptron (HMC-LMLP)

HMC-LMLP was presented by Cerri et al. [3] for solving hierarchical multi-label classification (HMC) to predict protein function. It showed that an accuracy can be significantly improved by sharing the output from an upper level with the lower level to predict classes. The model consists of many layers of Multi-Layer Perceptron (MLP) which the output from an upper level will be concatenated with a real feature. Then, the concatenated feature is used as a feature in the lower level as shown in Fig. 2.

Normally, when classifiers per level approach is applied for HMC task, the classifiers are separately trained and does not use any information from other classifiers. Therefore, some prediction from each classifier may be inconsistent. In HMC-LMLP, sharing the output from an upper level classifier with the lower level classifier is helped in this situation, because there is shared information between the upper classifier to the lower classifier.

D. Shared Hidden Layer Neural Network (SHL-NN)

According to [4], SHL-NN is the enhanced version of HMC-LMLP, which was designed for HTC. In SHL-NN, the shared information is not a raw output from the output layer of an upper level, but it is an output from the hidden layer of an upper level as shown in Fig. 3. The model can solve an issue of too large size of features, which often occurs in HTC and overwhelms the document embedding features when they are concatenated together.

However, the number of hidden nodes in each local classifier in SHL-NN must be limited and cannot be too large since it can outnumber the document embedding features. Therefore, the level with a lot of output classes will confront the limited accuracy.

III. Methodology

The workflow of an overall process from raw documents to their predictions is shown in Fig. 4. First, DocTag2Vec is employed to convert from a raw document to an embedding vector. In the training process of DocTag2Vec, there are two proposed embedding strategies to choose suitable labels from the class hierarchy to train the embedding model as shown in Section III (A). Then, the embedding vectors of training documents are fed to the proposed model called "Encoded Shared Layer Neural Network (ESL-NN)," which is described in Section III (B). Finally, a label correction is applied to fix an inconsistence prediction in terms of hierarchy.

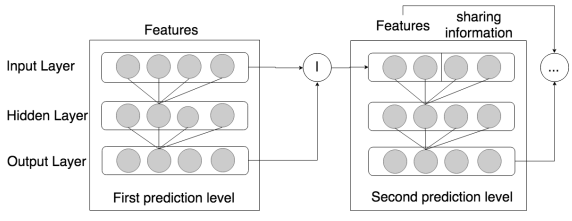

Fig. 2. The model architecture of HMC-LMLP

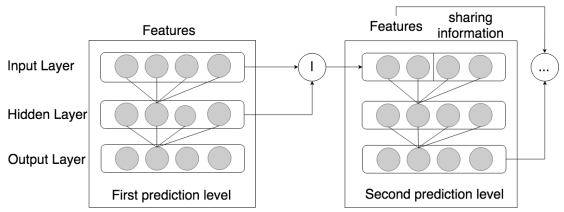

Fig. 3. The model architecture of SHL-NN

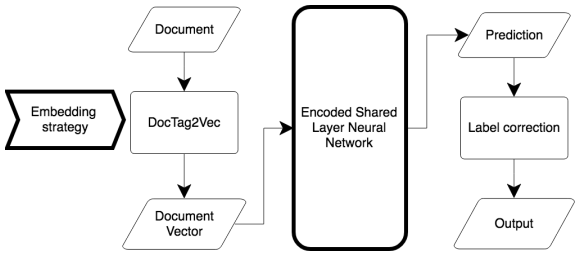

Fig. 4. Work flow of the proposed model

A. Embedding Strategies

In the training DocTag2Vec process, we must select a suitable class information from the hierarchy to supervise the model. There are two proposed strategies as shown below:

1) Leaf only strategy: It selects only leaf classes that is assigned to the document, while all non-leaf nodes are not used. The abbreviation of DocTag2Vec using Leaf only strategy is called "LOD-Vec."

2) Overall path strategy: It selects all classes that assigned to the document. If a document belongs to a child class, its ancester classes are also included in the training process. The abbreviation of DocTag2Vec using Overall path strategy is called "OPD-Vec."

B. Encoded Shared Layer Neural Network (ESL-NN)

The proposed architecture is shown in Fig. 5. The model is developed from HMC-LMLP like our previous work, SHL-NN. An output of the first level prediction is produced from a multi-layer perceptron with a document embedding vector. An output is encoded by using a layer of perceptron before it concatenates with a document embedding vector to be a feature to predict classes in the next level. It is different from SHL-NN that uses the output from hidden layers as shared information whose disadvantage is the number of hidden nodes will be limited because if the model has too many hidden nodes, the shared information from SHL-NN will overwhelm the document embedding feature. On the other hand, ESL-NN can the number of hidden nodes in the previous level classifier does not affect to the architecture of the classifier in the next level.

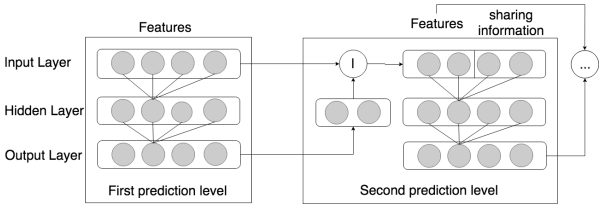

Fig. 5. The model architecture of ESL-NN

IV. DATASETS AND EXPERIMENTAL SETUP

A. Datasets

There are three datasets conducted using 5-folds stratified cross validation [10]. Table I and Table II show that the lower level in the hierarchy contains a large number of classes with insufficient training documents; therefore, we can only employ the stratification method on moderate appearance classes.

1) WIPO-alpha: The It is English patent collection of documents, which is used for automated categorization [11]. The structure of its hierarchy is tree, whose documents belong to only one leaf node. The database separates in 8 sections from A to H. We selected C and D-sections which are the largest and smallest sections, respectively. The text are only extracted from title and abstract of each document.

2) Wiki-small: It is a part of Wikipedia medium-sized dataset, which is an online encyclopedia created by the open community. Moreover, it is one of the challenging datasets in the the Large Sccale Hierarchical Text Classification Challenge (LSHTC3). Its class hierarhichy is graph. Features are extracted from contents in each page of Wikipedia [12].

B. Experimental Setups

All datasets are preprocessed using the Lancaster stemmer and removed stop words. Then, all words in each document are converted to embedding vectors using different strategies: Word2Vec, Doc2Vec and DocTag2Vec.

From hyperparameter tuning in DocTag2Vec, distributed bag of words (DBOW) is employed in the training process. The dimension of the document embedding is 150 dimensions with window size of 8 words. In the experiment on document embedding strategies, Doc2Vec is considered as a baseline strategy.

TABLE I. OVERALL DATA STATISTICS

Dataset	Class[a]	Depth	Instance	|Class|[b]
WIPO-D	832	4	1710	4
WIPO-C	5666	4	16245	4
Wiki-small	3238	4	60821	7.275

[a] Number of classes of all level.
[b] |Class| refers to average number of classes assigned to each document.

TABLE II. DATA STATISTIC OF EACH LEVEL

Level	WIPO-D		WIPO-C		Wiki-small	
	Class	*|Doc|*[a]	*Class*	*|Doc|*	*Class*	*|Doc|*
1	7	175.86	17	688.06	36	3785.61
2	20	61.55	56	208.88	562	309.70
3	160	7.69	852	13.73	2110	88.14
4	645	1.91	4741	2.48	3768	37.29

[a] |Doc| refers to average number of documents which belong to each class.

In SHL-NN (baseline) and ESL-NN (ours), the number of hidden nodes is set to be twice the number of classes in that level, but it must not over 300 and 3,000 units for SHL-NN and ESL-NN, respectively. The dropout rate of an input layer is set to 0.15, and the dropout rate after hidden layers is set to 0.25. In encoded shared layer of ESL-NN, the number of hidden nodes is set to be the number of classes of the previous level with the maximum size is 100 units and the dropout rate is 0.15. In the last layer of each level classifier, the sigmoid function is chosen to be an activation function. Meanwhile, other layers used ReLU function to be an activation function.

For an optimization, ADAM [13] is chosen with learning rate 0.001 and decay rate 0.01 using the multi-label soft margin loss as a loss function.

In the prediction process, the cut-off threshold is applied to decide which instance is positive or negative. It can be obtained by searching the one that reached the highest F1-macro score in validation set. Finally, the child-based correction must be employed to fix the inconsistence prediction in the class hierarchy.

took responsibility for deciding which instance is positive or negative. We chose cut-off threshold by searching the one that reached the highest F1-macro score in validation set. After that we selected child-based correction to fix the inconsistence prediction in term of hierarchy.

C. Evaluations

We used two evaluation metrics to compare each model, F-measures with macro- and micro-averaging for a multi-label purpose. They are defined as the performance criteria as shown in Table III. The macro-averaging is appropriate for measurement when we want to ignore the overcoming of majority classes. In opposition, micro-averaging is used to measure its globally counts.

A comparison in HTC often use F1 macro to be the main evaluation metric. For example, the LSHTC challenge on Kaggle, the competition in HTC, also used F1 macro to be a criterion for finding the winner of the competition.

V. Experimental results

We performed three experiments. First, we intent to find the best document embedding strategies for training document embedding. Second, we compared supervised document embedding and unsupervised sequence of word embedding which representation performs better with a same model. Finally, we compared ESL-NN with SHL-NN which have a same input. After that, the comparison of our proposed method, existing methods and baseline methods was made. The detail of experiments is described as below.

A. Embedding strategies

In the experiment, we used SHL-NN with different input features which are LOD-Vec, OPD-Vec and a baseline Doc2Vec to compare whose strategy for training a document embedding which one gives the best performance.

As shown in Table IV, we found that both features, LOD-Vec and OPD-Vec is better than the baseline Doc2Vec. At the same time, Overall path strategy which is a strategy for training OPD-Vec can achieved the highest accuracy in term of both F1 macro and F1 micro on every dataset. In term of F1 macro, OPD-Vec can achieve 6.93% higher average accuracy on each dataset than LOD-Vec and 55.42% higher than Doc2Vec. Moreover, in term of F1 micro, OPD-Vec can achieve 4.36% higher than LOD-Vec and 31.27% higher than Doc2Vec.

Moreover, we observed more about if the performance of OPD-Vec, which used all classes for training, can be affected by too many classes which are assigned in each document. A result on Wiki-small, whose average number of classes in each document is high, shows that OPD-Vec doesn't perform worse than LOD-Vec and Doc2Vec.

TABLE IV. Results of each document embedding strategy. Doc2Vec is baseline, where LOD-Vec and OPD-Vec are our approach.

Dataset	Document Embedding strategy	F1 macro	F1 micro
WIPO-D	Doc2Vec	0.0499	0.2748
	LOD-Vec	0.0709	0.3996
	OPD-Vec	**0.0725**	**0.4089**
WIPO-C	Doc2Vec	0.0214	0.3186
	LOD-Vec	0.0311	0.3682
	OPD-Vec	**0.0355**	**0.4008**
Wiki-small	Doc2Vec	0.1296	0.2754
	LOD-Vec	0.1822	0.3010
	OPD-Vec	**0.2069**	**0.3065**

[a] Bold face result is a winner on that dataset.

TABLE III. Evaluation metrics

Metric	Single-Label Classification	Multi-Label Classification	
		Macro-average	Micro-average
Precision	$Pr = TP / (TP + FP)$	$MaPr = \frac{1}{\lvert C\rvert}\sum_{i=1}^{\lvert C\rvert} Pr_i$	$MiPr = \frac{1}{\lvert C\rvert}\sum_{i=1}^{\lvert C\rvert} TP_i / \frac{1}{\lvert C\rvert}\sum_{i=1}^{\lvert C\rvert}(TP_i + FP_i)$
Recall	$Re = TP / (TP + FN)$	$MaRe = \frac{1}{\lvert C\rvert}\sum_{i=1}^{\lvert C\rvert} Re_i$	$MiRe = \frac{1}{\lvert C\rvert}\sum_{i=1}^{\lvert C\rvert} TP_i / \frac{1}{\lvert C\rvert}\sum_{i=1}^{\lvert C\rvert}(TP_i + FN_i)$
F_β	$F_\beta = \frac{(\beta^2+1) \times Pr \times Re}{\beta^2 + Pr + Re}$	$MaF_\beta = \frac{1}{\lvert C\rvert}\sum_{i=1}^{\lvert C\rvert} F_{\beta,i}$	$MiF_\beta = \frac{(\beta^2+1) \times MiPr \times MiRe}{\beta^2 + MiPr + MiRe}$

TABLE V. RESULTS OF EACH MODEL. THERE ARE TWO BASELINES: (I) SHL-NN+WORD2VEC AND (II) HR-SVM.

Dataset	Score	ESL-NN	SHL-NN + OPD-Vec	SHL-NN + Word2Vec	HR-SVM
WIPO-D	F1 macro	**0.0836**	0.0725	0.0293	0.0572
	F1 micro	**0.4512**	0.4089	0.3943	0.2085
WIPO-C	F1 macro	**0.0400**	0.0355	0.0096	0.0317
	F1 micro	0.4052	0.4008	**0.4128**	0.1075
Wiki-small	F1 macro	**0.2386**	0.2069	0.1121	0.2327
	F1 micro	0.3576	0.3065	**0.5763**	0.2200

[a.] Bold face result is a winner on that dataset.

TABLE VI. PERCENTAGE IMPROVEMENT FROM ORIGINAL METHODS TO PROPOSED METHODS

	SHL-NN + OPD-Vec	SHL-NN + Word2Vec	HR-SVM
ESL-NN	14.53 (3)	204.51 (3)	25.03 (3)
SHL-NN + OPD-Vec	0.00 (3)	166.59 (3)	9.18 (2)

[a.] In parentheses, we show the number of dataset that the proposed methods can achieved higher F1 macro than the original method significantly

B. Comparison between supervised document embedding and unsupervised sequence of word embedding.

This experiment intended to test Supervised document embedding and unsupervised sequence of word embedding which one performs better in both evaluation metric. We chose OPD-Vec which is the feature that give the best accuracy like we discussed in a previous section as supervised document embedding and chose a Word2Vec combined with CNN as unsupervised word embedding. In the experiment, SHL-NN was the model, which used for prediction with different features. The result of OPD-Vec and Word2Vec is shown in Table V.

The result in Table VI shows that OPD-Vec can achieve 166.59% higher accuracy in term of F1 macro. However, in term of F1 micro, OPD-Vec gets lower accuracy in WIPO-C and Wiki-small. As shown in Table VII, OPD-Vec is suitable for sparse label classification because it can achieve higher F1-macro in the level which have sparse labels. For instance, in the fourth level of every dataset, which always have a low average number of documents which belong to each class, OPD-Vec performs better compared with a sequence of Word2Vec. Because most of classes is always in level 4, so that is the reason why OPD-Vec can achieve extremely higher in term of F1 macro.

C. Comparison between SHL-NN and ESL-NN

This experiment compared SHL-NN and ESL-NN where their input is OPD-Vec. From the experimental result as shown in Table V, ESL-NN is get higher in both F1-micro and F1-macro. As shown in Table VI, in term of F1 macro, ESL-NN achieves 14.53% higher than SHL-NN. Moreover ESL-NN's F1 micro is 9.37% higher than SHL-NN.

As we saw more closely on the result of levels which have many classes, ESL-NN can overcome SHL-NN, especially on the fourth level of wiki-small which gets 21% higher accuracy. The reason is the number of hidden nodes of SHL-NN is limited because SHL-NN uses the output from hidden layers to be shared information. That causes the models whose levels have many classes will be too simple for predictions. Therefore, ESL-NN whose number of hidden nodes don't be limited for sharing information can perform better in the levels whose number of classes is high because the complexity of model in each level can be adjusted according to the number of classes on each level.

D. Existing method comparison

From the experiment before, we presented ESL-NN with OPD-Vec reaches the highest accuracy. Then we compared it with HR-SVM used TF-IDF features as an input and SHL-NN with sequence of word embedding which are existing methods.

Our purposed method can achieve the best accuracy on WIPO-D dataset. On the other hand, the proposed method can perform well only in term of F1-macro on WIPO-C and Wiki-small. In term of F1 macro, ESL-NN's F1-macro is 204.51% and 25.03% higher than SHL-NN and HR-SVM respectively as shown in Table VI. Moreover, ESL-NN can achieve the highest F1-macro significantly in every dataset with p-value less than 0.05 in a paired t-test.

As shown in Table VI, our proposed method gets a higher accuracy than existing methods on the lowest level on Wiki-small. Therefore, the result shows that ESL-NN can predict the specific class better than other existing models.

VI. CONCLUSION

In this paper, we aim to propose a novel deep learning network for hierarchical text categorization called "Encoded Shared Layer Neural Network (ESL-NN)." It is an extension of our previous work called "Shared Hidden Layer Neural Network (SHL-NN)." There are two main contributions in this work: (*i*) a supervised document embedding strategy and (*ii*) a novel sharing information technique. The experiment was conducted

TABLE VII. RESULT OF EACH MODEL ON EACH LEVEL IN TERMS OF F1 MACRO.

Dataset	Level	ESL-NN	SHL-NN + OPD-Vec	SHL-NN + Word2Vec	HR-SVM
WIPO-D	1	**0.6056**	0.4662	0.4583	0.2274
	2	**0.5249**	0.5311	0.3816	0.3469
	3	**0.1479**	0.1457	0.0601	0.1123
	4	**0.0483**	0.0359	0.0061	0.0276
WIPO-C	1	**0.6081**	0.6109	0.5002	0.1003
	2	**0.4552**	0.4548	0.2975	0.1947
	3	**0.0991**	0.0953	0.0219	0.0857
	4	**0.0225**	0.0177	0.0021	0.0191
Wiki-small	1	0.2700	0.2403	**0.4584**	0.1490
	2	0.2100	0.2157	**0.3842**	0.2793
	3	0.2445	0.2205	0.0574	**0.2620**
	4	**0.2392**	0.1977	0.0054	0.2081

[a.] Numbers which shown in the table is F1 macro.
[b.] Bold face result is a winner on that dataset.

on three corpuses: WIPO-C, WIPO-D, and Wiki-small. The macro-F1 results show that ESL-NN unanimously outperforms all baselines: SHL-NN and HR-SVM for 204.54% and 25.03%, consecutively. Among all the proposed document embedding strategies, the best approach is the DocTag2Vec using Overall path strategy (OPD-Vec). Furthermore, ESL-NN always shows better macro-F1 in the lower levels of the hierarchy; this illustrates that the proposed sharing strategy in ESL-NN is more efficient than the one in SHL-NN.

ACKNOWLEDGMENT

The authors gratefully acknowledge the financial supports from the Thailand Research Fund (TRF) under TRF Grant for New Researcher Scholar, Contact No. TRG5780220 and the Commission of Higher Education, Ministry of Education, Thailand.

REFERENCE

[1] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space," arXiv preprint arXiv:1301.3781, 2013.

[2] Y. Kim, "Convolutional neural networks for sentence classification," arXiv preprint arXiv:1408.5882, 2014.

[3] R. Cerri, R. C. Barros, P. L. F. a. A. C. de Carvalho and Y. Jin, "Reduction strategies for hierarchical multi-label classification in protein function prediction," BMC bioinformatics, vol. 17, no. 1, p. 373, 2016.

[4] M. Klungpornkun and P. Vateekul, "Hierarchical Text Categorization Using Level Based Neural Networks of Word Embedding Sequences with Sharing Layer Information," Computer Science and Software Engineering (JCSSE), 2017 14th International Joint Conference on. IEEE, 2017.

[5] P. Vateekul, M. Kubat and K. Sarinnapakorn, "Top-down optimized SVMs for hierarchical multi-label classification: A case study in gene function prediction," Intelligent Data Analysis.

[6] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," ACM Transactions on Intelligent Systems and Technology (TIST), vol. 2, p. 27, 2011.

[7] S. a. C. N. Jr and A. A. Freitas, "A survey of hierarchical classification across different application domains," Data Mining and Knowledge Discovery, vol. 22, pp. 31-72, 2011.

[8] Q. Le and T. Mikolov "Distributed representations of sentences and documents" In International Conference on Machine Learning, pp. 1188-1196.

[9] S. Chen, A. Soni, A. Pappu, and Y. Mehdad, "Doctag2vec: An embedding based multi-label learning approach for document tagging." arXiv preprint arXiv:1707.04596.

[10] K. Sechidis, G. Tsoumakas and I. Vlahavas, "On the stratification of multi-label data," Machine Learning and Knowledge Discovery in Databases, pp. 145-158, 2011.

[11] D. Tikk, G. Biró and J. D. Yang, "Experiment with a hierarchical text categorization method on WIPO patent collections," in Applied Research in Uncertainty Modeling and Analysis, Springer, 2005, pp. 283-302.

[12] I. Partalas, A. Kosmopoulos, N. Baskiotis, T. Artieres, G. Paliouras, E. Gaussier, I. Androutsopoulos, M.-R. Amini and P. Galinari, "LSHTC: A benchmark for large-scale text classification," arXiv preprint arXiv:1503.08581, 2015.

[13] DP. Kingma and J. Ba "Adam: A method for stochastic optimization" arXiv preprint arXiv:1412.6980, 2014.