Towards Automatic Coding of Collaborative Learning Data with Deep Learning Technology

Chihiro Shibata
School of Computer Sciences
Tokyo University of Technology
Tokyo, Japan
email:shibatachh@stf.teu.ac.jp

Kimihiko Ando
Cloud Service Center
Tokyo University of Technology
Tokyo, Japan
email:ando@stf.teu.ac.jp

Taketoshi Inaba
Graduate School of Bionics, Computer and Media Sciences
Tokyo University of Technology
Tokyo, Japan
email:inaba@stf.teu.ac.jp

***Abstract*— In Computer Supported Collaborative Learning (CSCL) research, gaining a guideline to carry out appropriate scaffolding by analyzing mechanism of successful collaborative interaction and extracting indicators to identify groups where collaborative process is not going well, can be considered as the most important preoccupation, both for research and for educational implementation. And to study this collaborative learning process, different approaches have been tried. In this paper, we opt for the verbal data analysis; its advantage of this method is that it enables quantitative processing while maintaining qualitative perspective, with collaborative learning data of considerable size. However, coding large scale educational data is extremely time consuming and sometimes goes beyond men's capacity. So, in recent years, there have also been attempts to automate complex coding by using machine learning technology. In this background, with large scale data generated in our CSCL system, we have tried to implement automation of high precision coding utilizing deep learning methods, which are derived from the leading edge technology of machine learning. The results indicate that our approach with deep learning methods is promising, outperforming the machine learning baselines, and that the prediction accuracy could be improved by constructing models more sensitive to the context of conversation.**

Keywords-CSCL; leaning analytics; coding scheme; deep learning methods.

I. INTRODUCTION

A. Analysis of collaborative process

One of the greatest research interests in the actual Computer Supported Collaborative Learning (CSCL) research is to analyze its social process from a social constructionist viewpoint, and key research questions are as follows: how knowledge and meanings are shared within a group, what types of conflict, synchronization and adjustment of opinions occur, and how knowledge is constructed from discussions. And answering to these questions enables to develop more effective scaffolding methods and CSCL system and tools.

In earlier researches at initial stage of CSCL, the focus was on each individual within a collaborating group, and the main point of interest had been how significantly a personal learning outcome was affected by characteristic types of a group (such as group size, group composition, learning tasks, and communication media) [1]. However, it gradually became clear that those characteristics are complexly connected and intertwined with each other, and showing causal relation to a specific result was extremely difficult. From the 1990s, the interest in CSCL research had moved away from awareness of the issue on how a personal learning is established within a group, to attempting to explain the process by clarifying the details of group interactions when learning is taking place within a group [2].

However, attempting to analyze collaborative process goes beyond merely shifting a research perspective; it also leads to fundamental re-examination of its analytical methodology. In other words, this involves a shift from quantitative analysis to qualitative analysis. Naturally, there are useful data among quantitative data saved within CSCL system, such as the number of contributions within a group, the number of contributions by each group member, and in some cases contribution attributes obtained from system interface (sentence opener), but those are very much a mere surface data. The most important data for analysis are contributions in chats, images/sounds within tools such as Skype, and various outputs generated in the process of collaborative learning; for analysis of those, ethnomethodologies such as conversation analysis and video analysis have been invoked [3] [4].

However, those researches by their very nature tend to be in-depth case studies of collaborative activities with a limited number of groups and have the disadvantage of not at all being easy to derive a guideline that has a certain level of universality and can be applicable in other contexts.

Therefore, researches have been carried out using verbal data analysis method that carry out coding from a perspective of linguistic or collaborative learning activities on a certain volume of language data generated in collaborative learning and analyzing them [5][6][7]. The advantage of this method is that it enables quantitative processing while maintaining qualitative perspective, with collaborative learning data of considerable size as the subject, while coding them manually is an extremely time consuming task which goes sometimes beyond men's capacity. For example, Persico et al. developed a technological tool which helps the tutors to code the contributions in chats and displays quantitative information about the qualitative information and coding data [8]. However, given that the coding procedure itself remains manual in most existing studies [9][10], there is an insurmountable limit in front of big data. Hence, we seek an automatic coding technique for a large scale collaborative learning data with deep learning methods.

B. Educational data and Learning Analytics

With the progress of educational cloud implementation in educational institutions, data generated in Learning Management System (LMS), e-learning, Social Network Service (SNS), Massive Open Online Course (MOOC) and others are increasing rapidly, and a new research approach called Learning Analytics (LA) that tries to gain knowledge that would lead to support of learning and educational activities by analyzing those educational big data is becoming more active [11][12]. Big educational data obtained from CSCL system integrated in educational cloud at a campus, such as conversation data, submitted documents and images/sounds of learning activities, will certainly become a subject for analysis in the near future: therefore, it is believed that we are coming into a time when it is necessary to seriously examine a new possibility of collaborative learning research as LA. Due to such background, in this research we have reconstructed CSCL system that has been operating in a campus server for the last five years as a module within Moodle, which is a LMS within the campus cloud, and have already structured an environment that can be operated within the campus and collect/analyze collaborative learning data.

C. The goal and purpose of this study

The goal of our research is to analyze large-scale collaborative data from LA perspective as described above and discover the mechanism of activation and deactivation of collaborative activity process which could not be gained from micro level case studies up to now. Furthermore, this research, based on its results, aims to implement supports in authentic learning/educational contexts, such as real-time monitoring of collaborative process and scaffolding to groups that are not becoming activated.

In this paper, as the first step towards this goal, we present work in progress, which attempts to develop an automation technique for coding of chat data and verifies its accuracy. To be more specific, a substantial volume of chat data is coded manually, and has a part of that learnt as training data in deep learning methods, which are derived from the leading edge technologies for machine learning; afterwards, automatic coding of the raw data is carried out. For validation of accuracy, the effectiveness of using deep learning methods is assessed by comparing accuracy against Naive Bayes and Support Vector Machines, which are baselines of machine learning algorithm used in existing studies that carried out automatic coding by machine learning.

D. Structure of this paper

This paper is structured as follows. In Section II, we present the related work. The Section III describes our datasets and coding scheme. The approach with deep learning methods for automatic coding is discussed in Section IV. Then, our experiment and results from our evaluation are described in Section V. Section VI concludes the paper.

II. RELATED WORK

Since deep learning can often outperform existing machine learning methods, such as SVMs, it has been applied in various research areas, such as image recognition and natural language processing [13]. Text classification is an important task in natural learning processing, for which various deep learning methods have been exploited extensively in recent studies. A structure called a CNN has been applied for text classification using word- or character-level modeling [14][15]. LSTM [16] and gated recurrent units (GRUs) [17] are popular structures for RNNs. Both structures are known to outperform existing models, such as n-grams, and are thus widely available as learning models for sequential data like text. RNNs are also applied to text classification in various ways [18][19]. For instance, Yang et al. used a bidirectional GRU with attention modeling by setting two hierarchical layers that consist of the word and sentence encoders [18].

In the field of CSCL, some researchers have tried to apply text classification technology to chat logs. The most representative studies would be Rosé and her colleagues'works [20][21][22]. For example, they applied text classification technology to a relatively large CSCL corpus that had been coded by human coders using the coding scheme with 7 dimensions, developed by Weinbeger and Fisher [21][23]. McLaren's Argunaut project took a similar approach: he used online discussions coded manually to train machine-learning classifiers in order to predict the appearance of these discussions characteristics in the new e-discussion[24]. However, it should be pointed out that all these prior studies rely on the machine learning techniques before deep learning studies emerge.

III. DATA AND CODING SCHEME

In this section, we explain how we collected our dataset and what coding scheme we adopted to categorize the dataset.

Copyright (c) IARIA, 2017. ISBN: 978-1-61208-541-8

A. Data Description

Our dataset obtained through chat function within the system, comes from conversations among students while carrying out online collaborative learning in university lectures using CSCL, which had been previously developed by the researchers of this study [25].

This CSCL is used without face to face contact; therefore, these data are all from occasions when unacquainted and separated students formed groups within lecture halls at the campus. And within the system all names of students are shown in nicknames, so that even if students knew each other they would not recognize each other.

The overview of CSCL contributions data used in this research is shown in Table 1. The number of lectures is seven and all classes of these lectures form groups of three to four; in fact, there are a lot of data that we could not process by coding them in this research. Learning times vary depending on the class, from 45 to 90 minutes. In total, the dataset contains 11504 contributions; there are 202 groups from all the classes, with 426 participating students; since students attend multiple classes, the number of participating students are smaller than the product of number of groups and number of students in a group.

Table 2 shows a conversation example of chat. This is a conversation example of three students.

TABLE I. CONTRIBUTIONS DATA USED IN THIS STUDY

Number of Lectures	7 Lectures
Member of Groups	3-4 people
Learning Time	45-90 mintutes
Number of Groups	202 groups
Number of Students	426 students

TABLE II. CONVERSATION EXAMPLE (TRANLATION FROM JAPANESE)

Talker	Contents
D	Where do you want to change?
E	That's right … I guess, first of all, we definitely need to change the question, and then, what about the well-formed formula?
D	How is it that changes only the third line of the question?
D	Regarding the well-formed formula, it's the final part after ⊃.
E	That's good idea.
F	I agree. How do we want to change that?

B. Coding scheme

In accordance with our manual for code assignment, one code label is assigned to one contribution in a chat. There are 16 types of code labels as shown in Table 3, and one of those labels is assigned for all cases.

All labels in our dataset are coded by two people; the coincidence rate between the labels assigned was 67%. However, when we reviewed the resultant coding data, it was discovered that there were duplicated labels for some contributions, and some labels had variances depending on the coder; therefore, after conferring among us, we unified labels and re-coded the contributions. The resultant number of labels assigned is shown in Table 3. Concordance rate is 82.3% and this is a high concordance rate with 0.800 Kappa coefficient, and we consider this to be sufficiently practical for use as an educational dataset in deep learning methods. Fig. 1 shows the frequencies of the labels in the dataset. Nine labels describe more than 90% of occurrences; label occurrences appear to have a long-tail distribution. The main purpose of this study is to learn and infer these labels from posted contributions.

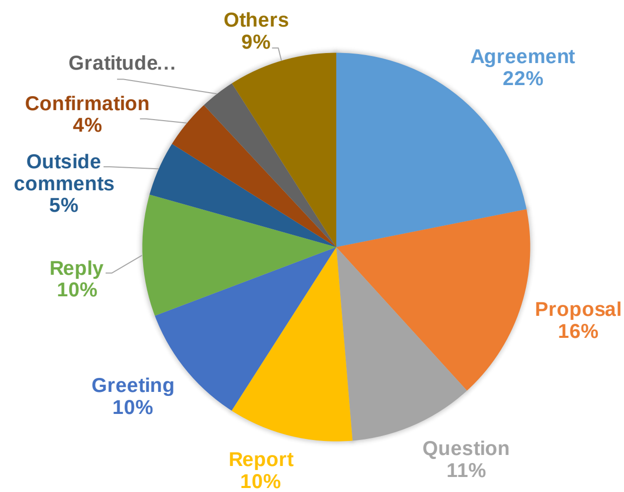

Figure 1. Ratio of each conversational coding labels

IV. APPROACH -- DEEP LEARNING

In recent years, deep learning technology has led to dramatic developments in the field of artificial intelligence. Deep learning is a general framework of learning methods that use neural networks with millions of weight parameters. The weights in neural networks are optimized so that their output coincides with labels in the given data. With the recent development of parallel computing using Graphics Processing Units (GPUs) and optimization algorithms, machines are able to learn large numbers of parameters from large datasets at realistic costs.

To try automatic coding, we adapt three types of deep neural network (DNN) structures: a convolutional neural network (CNN) -based model and two bidirectional Long short-term memory (LSTM) -based models, LSTM and Sequence-to-Sequence (Seq2Seq). The first and second models take only a single contribution as input and cannot refer to context information in the conversation. Conversely, the Seq2Seq model can capture context information by using a pair of sentences as its input, which represent source and replay contributions.

A. CNN-based model

The CNN-based model uses the network architecture proposed by Kim et al. (Fig. 2). Before training, all words in

TABLE III. List of labels

Tag	Meaning of tag	Contribution example	Number of times used
Agreement	Affirmative reply	I think that's good	5033
Proposal	Conveying opinion, or yes/no question	How about five of us here make the submission?	3762
Question	Other than yes/no question	What shall we do with the title?	2399
Report	Reporting own status	I corrected the complicated one	2394
Greeting	Greeting to other members	I'm looking forward to working with you	2342
Reply	Other replies	It looks that way!	2324
Outside comments	Contribution on matters other than assignment contents Opinions on systems and such	My contribution is disappearing already; so fast! A bug	1049
Confirmation	Confirm the assignment and how to proceed	Would you like to submit it now?	949
Gratitude	Gratitude to other members	Thanks!	671
Switchover	A contribution to change event being handled, such as moving on to the next assignment	Shall we give it a try?	625
Joke	Joke to other members	You should, like, learn it physically? :)	433
Request	Requesting somebody to do some task	Can either of you reply?	354
Correction	Correcting past contribution	Sorry, I meant children	204
Disagreement	Negative reply	I think 30 minute is too long	160
Complaint	Dissatisfactions towards assignments or systems	I must say the theme isn't great	155
Noise	Contribution that does not make sense	?meet? day???	143

the data are converted to word vectors. Word vectors are often obtained by pre-training using another external dataset. In this study, we implemented two types of word vectors: 1) vectors obtained by applying word2vec (the skipped gram model with negative sampling) to all Japanese text in Wikipedia, and 2) randomly initialized vectors that are tuned simultaneously with the CNN.

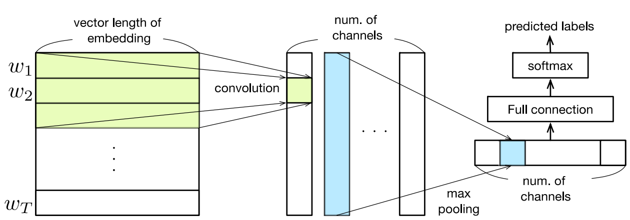

Figure 2. CNN-based model

B. Bidirectional LSTM-based model

An LSTM is a recurrent neural networks (RNNs) that is carefully constructed so that it can capture long-distance dependencies in sequential data. Generally speaking, an RNN consists of input vector x_t and output vector y_t for each time t. To obtain the output $y_{\{t\}}$, the previous output vector $y_{\{t-1\}}$ is fed to the neural network along with the current input vector x_t. The LSTM has another hidden vector, c_t, called the *state vector* in addition to the input and output vectors. While the state vector is also output from the neural network, it is computed to track long-distance relations through a function called a *forget gate*, which is designed to decide whether the state vector should be changed. We feed word vectors into the two-layer LSTM network sequentially in both the forward and reverse directions. After all words in a contribution are input, both output vectors are concatenated and fed into the two-layer fully-connected network and the softmax layer to obtain classification results. Fig. 3 illustrates this architecture.

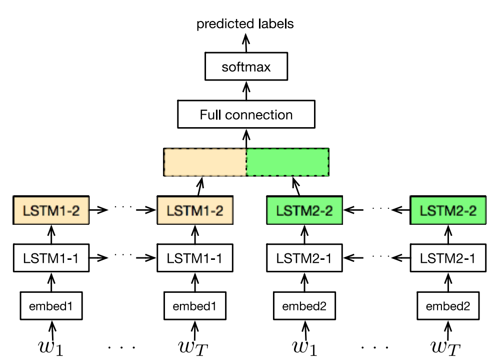

Figure 3. Bidirectional LSTM-based

C. Bidirectional Seq2Seq-based model

Each contribution is a part of a conversation; therefore, to classify labels more accurately, we must account for conversational contexts. To do this, we convert all contributions in conversations into pairs of *source* and *reply* contributions. Even if a user posts a contribution that does not explicitly cite another, we assume that it cites a previous contribution. We also suppose that the first contribution of each conversation cites the empty string. To construct a model that regards the source contribution as a conversational context and the reply as a representation of the user's intention, we use the Seq2seq framework. Seq2seq

Copyright (c) IARIA, 2017.

[26] was originally proposed as a neural model using RNNs for machine translation, and later applied to other tasks, such as conversational generation [27]. It consists of two separate LSTM networks, called the encoder and decoder. We use two-layer LSTM networks for both the encoder and decoder. Words are sequentially fed in both the forward and reverse directions. Output vectors from decoders are concatenated and fed into the two-layer fully-connected network and the softmax layer (Fig. 4).

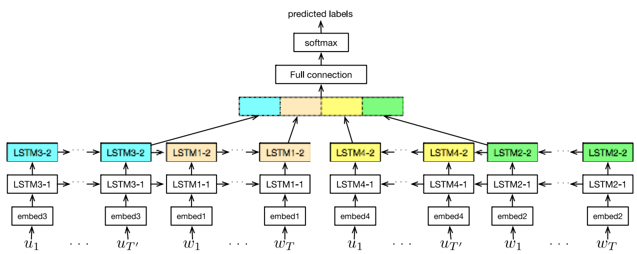

Figure 4. Bidirectional Seq2Seq-based model

V. EVALUATION

For each contribution, we trimmed sentences beginning with the symbol ">," which were automatically generated by the system. Since all the data consist of Japanese text, morphological analysis was needed. We split texts into words using a tool called MeCab. Replacing low-frequency words with "unknown," the vocabulary size was decreased to approximately 4,000. Each contribution was given two labels annotated by different people; we removed contributions that were assigned two different labels. We used 90% of the remaining 8,015 contributions as training data and 10% as test data. The accuracy of the learning result for each model is measured with the test data.

A. Baseline Methods

For comparison, we used three classifiers; Naive Bayes, a linear support vector machine (SVM), and an SVM with a radial basis function (RBF) kernel. We also used two types of feature sets: unigrams only and unigrams and bigrams. For the SVM classifiers, in order to improve the classification accuracy, input vectors were obtained by normalizing zero-one vectors whose elements represent occurrences of unigrams or bigrams.

B. Model Parameters and Learning

Model parameters, such as the vector sizes of layers, are determined as follows. Both the size of word embedding and the size of the last fully connected layer are 200 for all models. We set the patch size of the convolutional layer in the vertical direction to 4 and the number of channels to 256 for the CNN-based models. We set the size of both LSTM layers to 800 for the LSTM and Seq2Seq models.

Models are learned by stochastic descent gradient (SDG) using an optimization method called Adam. To avoid overfitting, iteration was stopped at 10 epochs for the LSTM-based methods and 30 epochs for the CNN-based methods. Due to the fluctuation in accuracy results between epochs, we took the average of the last 5 epochs to measure the accuracy of each model. To prevent overfitting, dropout was applied to the last and second-last fully connected layers.

C. Experimental Results

Table 4 shows the accuracies of the three DNN models and baseline methods. Overall, the DNN models outperform the baselines, even as the SVMs maintain their high performance. Among baseline methods, the SVM with the RBF kernel achieved the highest accuracy. For the CNN-based models, using word vectors trained using the Wikipedia data slightly enhanced accuracy. For LSTM-based models, bidirectional processing yielded slightly higher accuracy than single-directional processing.

There was no significant difference in the accuracies of the CNN model using Wikipedia and the bidirectional LSTM model. Both of these methods outperformed the best of SVMs by 1–2%.

Seq2Seq model outperformed other methods clearly; the best of SVMs by 5-6% and other DNN models by 3-4%.

TABLE IV. PREDICTIVE ACCURACIES FOR BASELINES AND DEEP-NEURAL-NETWORK MODELS

Naïve Bayes		SVM(Linear)		SVM(RBF Kernel)	
unigram	uni+bigram	unigram	uni+bigram	unigram	uni+bigram
0.554	0.598	0.642	0.659	**0.664**	0.659

CNN		LSTM		Seq2Seq	
with wikipedia	w.o. wikipedia	single-direction	bidirection	bidirection	bidir. w. interm.
0.686	0.677	0.676	0.678	**0.718**	0.717

The kappa coefficient for the bidirectional LSTM model was 0.63, which is sufficiently high. However, to automatically comprehend and judge the activities of users from only the labels inferred by machines, the kappa coefficient must be improved. By using the Seq2Seq model, which is able to capture the contextual information from the source or the adjacent contribution, the kappa coefficient was improved to 0.723.

Hereafter, we analyze the misclassification of each label individually. The precision and recall for each label are shown in Table 5. Of the ten most frequent labels, the precision of "Greeting" predictions were highest (F1: 0.94) and that of "Agreement" was the second highest (F1: 0.83). "Question" was also predicted with high accuracy (F1: 0.77). These results are consistent with our intuition, as both seem to be easy to infer from the contributions themselves, without knowing their context. In contrast, as Table 5 shows, the label "Reply" was hard for our model to predict. That performed worst with respect to the recall, tending to be misclassified as an "Agreement", "Proposal" or "Report," as shown in the confusion matrix (Fig. 5). This can be solved if richer context in neighboring contributions is used as input to classifiers in addition to the source contribution.

VI. CONCLUSION AND FUTURE WORK

As the first step to analyze collaborative process of big educational data, we tried to automate time-consuming

TABLE V. PRESITION AND RECALL FOR EACH LABEL (RESULT OF BI-DIRECTIONAL LSTM)

	Presition	Recall	F1-value
Agreement	0.85	0.81	0.83
Proposal	0.73	0.74	0.73
Question	0.75	0.8	0.77
Report	0.64	0.62	0.63
Greeting	0.94	0.94	0.94
Reply	0.62	0.46	0.53
Outside comments	0.17	0.47	0.25
Confirmation	0.58	0.74	0.65
Gratitude	0.67	0.67	0.67

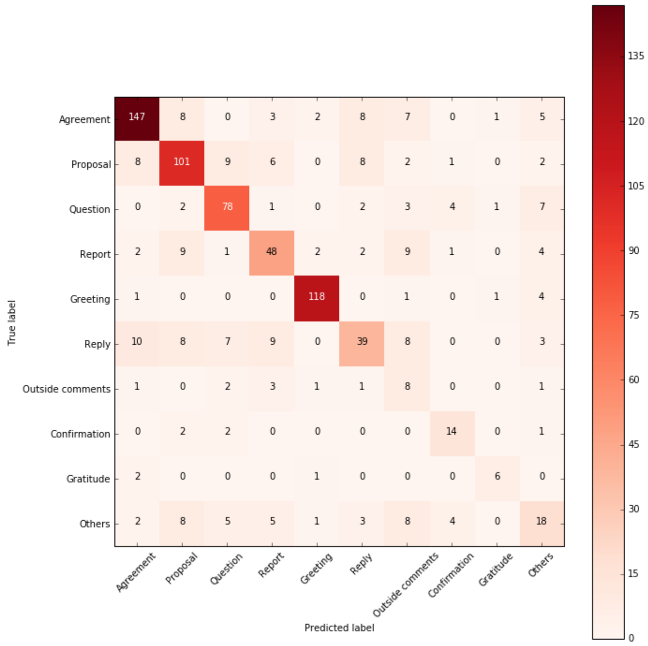

Figure-5. Confusion matrix for the Seq2S2q model.

coding task by using deep learning methods. The result was promising; our approach, particularly, Seq2Seq model outperformed other methods clearly; the best of SVMs by 5-6% and other DNN models by 3-4%. It seems that this model could obtain almost the same predictive accuracy with other coding schemes than ours, for the reason that our coding scheme is sufficiently complex with 16 labels, based not on the surface information, but on the contextual significance of each contribution.

As for the future research directions, we may have two approaches to pursue. The first approach concerns coding scheme. Our scheme, based on speech acts, was sufficiently complex, but not global. To capture the collaborative process more precisely, it will be necessary to construct a coding scheme which is more sensitive to details of interaciton and social cognitive process of learning. The second approach is about DNN models. To improve prediction accuracy, it may be effective to introduce an attention model to our DNN models. In addition, the context of conversation should be considered. To capture context more precisely, it may be necessary to construct more complex models that take multiple preceding contributions as input vectors.

ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI Grant Number 26350289 and 16K01134.

REFERENCES

[1] G. Stahl, T. Koschmann, and D. Suthers, “Computer-supported collaborative learning,” In The Cambridge handbook of the learning science, K. Sawyer, Eds. Cambridge university press, pp.479-500, 2014.

[2] P. Dillenbourg, P. Baker, A. Blaye, and C. O’Malley, “The evolution of research on collaborative learning,” In Learning in humans and machines: Towards an interdisciplinary learning science, P. Reimann and H. Spada, Eds. Oxford: Elservier, pp. 189-211, 1996.

[3] T. Koschmann, “Understanding understanding in action,” Journal of Pragmatics, 43, pp435-437, 2011.

[4] T. Koschmann, G. Stahl, and A.Zemel, “The video analyst’s manifesto (or The implications of Garfinkel’s policies for the development of a program of video analysis research within the learning science),” In Video reseach in the learning sciences, R. Goldman, R. Pea, B. Barron and S. Derry, Eds. Routledge, pp.133-144, 2007.

[5] M. Chi, “Quantifying qualitative analyses of verbal data : A pratical gauide ,” Journal of the Leanrnig Science, 6(3), pp.271-315, 1997.

[6] A. Meier, H. Spada, and N. Rummel, “A rating scheme for assesseing the quality of coputer-supported collaboration processes,” International Jounala of Computer Suppported Collaborative Learning, 2, pp.63-86, 2007.

[7] H. Jeong, “Verbal data analysis for understanding interacitons,” In The International Handbook of Collaborative Learning, C. Hmelo-Silver, A. M. O’Donnell, C. Chan and C. Chin, Eds. Routledge, pp.168-183, 2013.

[8] D. Persico, F. Pozzi, and L. Sarti, “Monitaring collaborative activities in computer supported learning,” Distance Education, 31(1), pp.5-22, 2010.

[9] L. Lipponen, M. Rahikainen, J. Lamillo, and K. Hakkarainen, “Patterns of participation and discourse in elementary students’computer-supported collaborative learning,”, Learning and Instruction, 13, pp.487-509, 2003.

[10] S. Schrire, “Knowledge building in asynchronous discussion groups: Going beyond quantitative analysis,” Computer & Education 46, pp.49-70, 2006.

[11] 1st Internationa Conference on Learning Analytics and Knowledge. [Online]. Avaiable from: https://tekri.athabascau.ca/analytics/ 2016.11.29

[12] B. R. Schaun and P. S. Inventado, “Educational data mining and learning analytics,” In Learning Analytics, J. A. Larusoon and B. White, Eds. Springer, pp.61-75, 2014.

[13] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, 521(7553), pp.436--444, 2015.

[14] Y. Kim, “Convolutional neural networks for sentence classification,” arXiv preprint arXiv:1408.5882, 2014.

[15] X. Zhang, J. Zhao, and Y.LeCun. “Character-level convolutional networks for text classification,” In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS2015), pp.649-657, 2015.

Copyright (c) IARIA, 2017.

[16] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, 9(8), pp.1735--1780, 1997.

[17] J. Chung, C. Gulcehre, K. Hyun Cho and Y. Bengio, “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling,” arXiv preprint arXiv:1412.3555, 2014.

[18] Z. Yang, et al., “Hierarchical Attention Networks for Document Classification,” In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics(NAACL2016), Human Language Technologies, 2016.

[19] D. Tang, B. Qin, and T. Liu, “Document modeling with gated recurrent neural network for sentiment classification,” In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing(EMNLP2016), pp.1422–1432, 2015.

[20] C. Rosé, et al., “Towards an interactive assessment framework for engineering design project based learning,” In Proceedings of DETC2007, 2007.

[21] C. Rosé, et al., “Analyzing collaborative learning processes automatically: Exploiting the advances of computational linguistics in computer-supported collaborative learning,” International Journal of Computer Supported Collaborative Learning, 3(3), pp.237-271, 2008.

[22] G. Gweon, S. Soojin, J. Lee, S. Finger and C.Rosé, “A framework for assessment of student project groups on-line and off-line,” In Analyzing Interactions in CSCL: Methods, Approaches and Issues, S. Putambekar, G.Erkens and C. Hmelo-Silver Eds. Springer, pp.293-317, 2011.

[23] A. Weinberger and F. Fischer, “A frame work to analyze arugmetative knowledge construciton in computer-supported learning,” Computer & Education, 46(1), pp.71-95, 2006.

[24] B. McLaren, O. Scheuer, M. De Laat, H. Hever and R. De Groot, “Using machine learning techniques to analysze and support mediation of student e-discussions,” In Proceedings of artificial intelligence in education, 2007.

[25] T. Inaba and K. Ando, “Development and Evaluation of CSCL System for Large Classrooms Using Question-Posing Script,” International Journal on Advances in Software, 7(3&4),pp.590-600, 2014.

[26] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” arXiv preprint arXiv, pp.1409.0473, 2014.

[27] O. Vinyals and Q. V. Le, “ A Neural Conversational Mode,” arXiv preprint arXiv:1506.05869, (ICML Deep Learning Workshop 2015), 2015.