Yang, J. C. et al. (Eds.) (2018). Proceedings of the 26th International Conference on Computers
in Education. Philippines: Asia-Pacific Society for Computers in Education

Identifying teamwork indicators in an online collaborative problem-solving task: A text-mining approach

Dhivya SURESH [a*], Hsiang Hui LEK[b], & Elizabeth KOH [a*]
[a]*National Institute of Education, Nanyang Technological University, Singapore*
[b]*National University of Singapore, Singapore*
[*]dhivya.suresh@nie.edu.sg

Abstract: Teamwork is an important competency for 21st century learner. However, equipping students with an awareness of their teamwork behaviors is difficult. This paper therefore aims to develop a model that will analyze student dialogue to identify teamwork indicators that will serve as formative feedback for students. Four dimensions of teamwork namely coordination, mutual performance monitoring, constructive conflict and team emotional support are measured. In addition, the paper explores multi-label classification approaches combined with feature engineering techniques to classify student chat data. The results show that by incorporating linguistic features, it is possible to achieve better performance in identifying the teamwork indicators in student dialogue.

Keywords: Teamwork, Pre-processing, Supervised machine learning, Text mining, Learning analytics, Feature engineering, Formative assessment, Chatlog

1. Introduction

Teamwork is an important competency for our 21st century learners. In collaborative problem-solving tasks, good teamwork helps in promoting task success. Moreover, teamwork is an important end in itself, as students will have to work in various teams as they progress in their life journey. However, students may be unaware of their behaviors and how it contributes to or harms the team. Therefore, there is a need for ways to provide students with greater awareness of their behaviors in teams. This research examines the text chatlog of students in an online chat while participating in an online collaborative problem-solving task. It aims to identify teamwork indicators from their dialogue in order to provide students with feedback to become more aware of their teamwork competency.

A text-mining approach is adopted in this paper with an elaborate effort in feature engineering to incorporate linguistic properties of the text into text classification. This approach is a refinement of a previous method by Shibani, Koh, Lai, & Shim (2017) which considered a rule-based classification and a simple unigram text classification. In this paper, we also performed a comparison study and the results show that by incorporating linguistic features, it is possible to achieve better performance in identifying the teamwork indicators. The paper focuses on four main teamwork dimensions – coordination, mutual performance monitoring, constructive conflict and team emotional support (Koh, Hong, & Tan, 2018), which were drawn from several empirical and conceptual studies. The dataset for the chatlog is taken from an in-school activity, where Secondary school students in teams of three and four, participated in a collaborative problem-solving activity. They had about 45 minutes to solve an icebreaker task followed by the main dilemma task.

This paper seeks to evaluate a refined approach using feature engineering to assess teamwork dialogue. The remaining sections of the paper are organized as follows: Section 2 is a brief literature review on teamwork and the various multi-label approaches in text data. Section 3 describes the methods and methodology. Section 4 discusses the results and analysis while section 5 concludes the paper with implications and future work.

2. Literature Review

Traditionally, assessment of teamwork projects focusses heavily on the application of knowledge and the outcome rather than the teamwork process (Hughes & Jones, 2011). Feedback to students tends to be towards the end rather than during the collaborative problem-solving process itself. Formative assessment is an important pedagogy to help students be more aware of their behaviors and understanding especially in teams (Strijbos & Sluijsmans, 2010). The feedback can serve as the basis for their change and possible transformation.

Several studies have used a text mining approach to provide students with indicators of their online behaviors. He (2003) for example, analyzed questions and chat messages posted online to examine the patterns in student learning behaviors in online platforms. While there has been some work done in extracting meaning from chatlog text (Anjewierden, Kolloffel, & Hulshof, 2007; Rosa & Ellen, 2009), there have been limited studies that focus on the topic of teamwork.

A promising approach to analyzing chat text is feature engineering and data pre-processing (Aggarwal & Zhai, 2012). Previous work (Chandrasekar & Qian, 2016; Günal, Ergin, Gülmezoğlu, & Gerek, 2006; Uysal & Gunal, 2014) have shown that feature engineering and data pre-processing play a significant part in the performance of text classification tasks. Some of the features that are typically used for this problem include part of speech (POS) tags, named entity, bag of words (BOW), and data pre-processing typically include tokenization, removal of stop words, and stemming (Uysal & Gunal, 2014).

Most of the work in text classification has considered single-label classification where each classification instance has only one class label. The commonly used machine learning algorithms that have been previously adopted include Naïve Bayes classifiers, Decision Tree classifiers, Support Vector Machines, Rule-based classifiers, and Neural Networks (See Allahyari et al. (2017) and Aggarwal & Zhai (2012) for a survey of some of these approaches). However, it is possible to perform multi-label classification where each classification instance can have one or more class labels. Gibaja and Ventura (2015) group such approaches into two main categories: problem transformation methods and algorithm adaption methods. Problem transformation methods generate multiple binary classifiers, one for each label and combine the classification results together in order to achieve multi-label classification. Whereas, algorithm adaption methods extend the single-label algorithm in order to directly deal with multi-label data. This approach may be suitable for identifying teamwork indicators from chat text.

3. Methods and methodology

This section covers the various approaches used for the classification task of our research project.

3.1 Teamwork competency dimensions

This project is part of a larger study on teamwork in which teamwork competency is conceptualized as a multi-dimensional concept of the process of members working in a team (Koh et al., 2018; Salas et al., 2009). Four dimensions of teamwork are focused on; these are applicable across team types and task (Salas et al., 2009). They are:

1. Coordination (COD) – the ability to organize team activity to the complete task on time
2. Mutual Performance Monitoring (MPM) – the ability to track the performance of team members
3. Constructive Conflict (CSC) – the ability to deal with differences in interpretation between team members through discussion and clarification.
4. Team Emotional Support (TES) – the ability to bond emotionally and provide psychological support to other team members

Two dimensions from previous work (Koh et al., 2018; Shibani et al., 2017) are excluded from this study due to theoretical, methodological and practical reasons including the importance of developing measures that are easy to understand and distinct, that would surface in sufficient quantities for analysis and enable easier future applications.

3.2 Corpus and Manual Coding of chat data

Based on the above-mentioned four dimensions, a coding scheme was created and used for manual coding. The corpus for this text analysis consists of 19,762 chat messages collected from 272 students in 76 teams who participated in the study. A subset of chat data from seven teams was annotated by two coders with a Cohen's kappa > 0.65 after which the entire corpus was coded individually by them. The unit of analysis was a chatline, and each chatlog could be annotated for any of the four dimensions, or had no code (nc) if the line did not fall into any dimension. In addition, the team found it useful to add a spam category, to mark the line as spam. Manually coded data from 76 teams excluding the spam and nc was used for the classification task. The chat data set was split for training and test purposes such that 80% of the data consisting of 9,893 lines formed the training set and 20% of the data consisting of 2,474 lines formed the test set. To better understand the context of the data, example chatlines of the four teamwork dimensions are shown in Table 1.

Table 1

Sample Coding for Teamwork Dimensions in a Chat Log

Name	**Message**	**COD**	**MPM**	**CSC**	**TES**
A	Hi	0	0	0	1
B	Alice's here?	1	0	0	0
A	Yep	1	0	0	0
Chat Admin	/takeover Lets do an ice-breaker activity! Describe your ideal teacher	0	0	0	0
A	an ideal teacher is someone that is understanding	0	0	1	0
A	u all no idea ah	0	1	0	0
B	alice describe cher sia	0	1	0	0

3.3 Pre-processing

To prepare the data for pre-processing, nc chatlines were removed from the dataset. The data preparation step was followed by data pre-processing to simplify the text such that the classifier could easily learn the features. Pre-processing steps include

1) Emotions and punctuation tagging: to replace all emotions and punctuation with tags
2) Chat abbreviation expansion: to expand short forms and acronyms
3) Local terms replacement: to replace all Singapore English terms with English equivalents
4) Named entity recognition: To replace all names with a NAME tag

The above-mentioned steps in preprocessing were done as part of previous research and a detailed description of each of the steps in preprocessing is described in Shibani et al. (2017). An example of each of the preprocessing steps is given in Table 2.

Table 2

Examples of the Pre-processing Steps

Step No	Preprocessing Step	Example
1	Emotion & Punctuation Tagging	! - ^exclaim_mark^ :) - ^pos_emo^
2	Chat Abbreviation Expansion	jk - just kidding, lol - laughing out loud
3	Local Terms Replacement	ah ma - grandma, can lah – okay
4	Named Entity Recognition	Alice - ^NAME^

3.4 Feature Engineering and Extraction

Features are the basic building block for machine learning algorithms. The features in the dataset have a huge impact in the machine learning outputs. The quality of features in dataset can be improved using processes such as feature engineering and feature selection. Our current research focusses more on the creation of new features to improve the performance of the classifiers. Ten new features were created by writing context-sensitive rules using indicative terms dictionary, POS tagging and regular expressions. These newly created features are then passed to the classifier along with the existing features. The lists of the ten new features created are listed in Table 3.

An example of how a feature was created is briefly explained. According to the coding scheme, chatlines that elaborate on an idea is coded as part of the category, constructive conflict. So a new feature named idea elaboration is created as follows

1. An indicative terms dictionary consisting of task-related words was created
2. Then the conditions for idea elaboration were created using the coding scheme
    a. Condition 1: Chatline should contain more than three task-related words from the indicative terms dictionary
    b. Condition 2: Chatline should not contain a <final_answer> tag as it does not representation elaboration of ideas
3. If both the conditions are satisfied then the F_ELABORATION feature is set to 1.

```
if condition 1 & condition 2:
        append Idea Elaboration tag
else:
        return chatline
```

This is followed by a unigram/bag of words feature extraction step. In order for the machine learning algorithms to use these features, a vectorization process adopted to convert their values into a numeric form. The end result of this process produces a vector where each unique word token is a column and each document is a row of numeric values for each of these unique token. For our current dataset, we experimented using three vectorizers (i) Count vectorizer (ii) Hash vectorizer (iii) term frequency-inverse document frequency (TF-IDF) vectorizer. A closer look at the result of vectorization showed that the TF-IDF vectorizer performed better as it chose the most important features rather than selecting word just based on the count of their occurrences.

Table 3

List of Features

Sno	Feature Name	Description	Example
1	F_Time	Identifies chatlines that express task related time comments	"5 more minutes" "Faster lah"
2	F_INSTRUCTION	Identifies chatlines that instruct team members to perform an activity and come up with answers	"Describe lah" "Combine the answer"
3	F_PROGRESS	Identifies chatlines which indicates student's sharing their progress on the activity	"Solved already" "We are done"
4	F_CLARIFICATIONQN	Identifies chatlines that are clarification questions related to the task	"Patient and Interesting?" "They should find a way to reduce the smoke produced?"
5	F_ELABOARATION	Identifies chatlines that elaborate on ideas	“I don’t think we can find a way to reduce the smoke produced” "At least your dad would be able to find another job"

6	F_DISAGREEMENT	Identifies chatlines that expresses student disagreement on ideas	"i disagree"
7	F_GREETING	Identifies chatlines that are greetings	"Hi", "Good morning guys"
8	F_POSEMO	Identifies chatlines that express positive emotions and humorous talks	"Yay!", "just kidding"
9	F_AGREEMENT	Identifies chatlines that express explicit agreement	"agree with that", "yes ok"
10	F_APPRECIATION	Identifies chatlines that express appreciation	"Alice types the fastest", "Good job Bob"

3.5 Classification

Traditional single-label classification tasks are typical supervised learning problems where each instance is associated with a single label learnt from a set of examples with a single label. Multi-label classification on the other hand, is different in a way where each instance is associated with multiple labels. There are multiple approaches to handle multi-label classification. Two common methods include problem transformation and algorithm adaption.

For the problem transformation approach, we experimented using the most widely used transformation method known as the Binary Relevance (BR Learning). It is the same as the one-versus-all (OVA) approach of solving a multi-class problem using a binary classifier (Katakis, Tsoumakas, & Vlahavas, 2008). To train the classifier we used the Support Vector Machine (SVM) learning algorithm with a linear kernel.

With regard to the adaptive algorithm approach, we worked with five different classifiers to evaluate which one was the best in classifying our chat dataset. These classifiers include KNeighborsClassifier (KNN), DecisionTreeClassifier (DT), ExtraTreeClassifier (ET), ExtraTreesClassifier (ETs) and RandomForestClassifier (RT).

3.6 Evaluation Measures

The evaluation measures for multi-label classification are different in contrast to the traditional single label classification. While some of the evaluation measures are carried out on a per-label basis, other measures are based on evaluation of label sets. The former is called label based evaluation and the latter is called label-set based evaluation (Read, Pfahringer, Holmes, & Frank, 2011).

In order to compare the performance of the classifiers, the following metrics were calculated: Precision, Recall, F Score, Accuracy and Hamming Loss. In multi-label context, accuracy is a label-set measure where the set of labels predicted for a sample must exactly match the corresponding set of test labels. When a predicted set of labels exactly matches the true set of labels, the evaluation is called the exact match measure or 0/1 loss. However, 0/1 loss tends to be too strict as the entire set of labels must be correctly predicted. Whereas Hamming Loss on the other hand based on the binary evaluation of each of the labels assigned. This measure is lenient when compared to the 0/1 loss.

Precision is the ratio of positive examples that were correctly classified to the total number examples labeled as positive. Recall is the ratio of number of correctly classified positive examples to the total number of positive examples in the data (Sokolova & Lapalme, 2009). F Score is the harmonic mean of precision and recall.

4. Results and Discussion

This section discusses the results based on three different comparisons to answer the following questions.

1. Can feature engineering lead to better classification performance?
2. Which approach, binary relevance or adaptive algorithm, performed better in assessing
3. teamwork dialogue?
4. Which classifier performed the best in classifying teamwork dimensions?

The results of the classification task are presented in terms of the following metrics: Precision, Recall, F Score, Accuracy and Hamming Loss. Results of the different classifiers were compared with two sets of data 1) The baseline with no new features added 2) The feature engineered dataset with additional new features. Table 4 reports the performance metrics of the six different classifiers that were used for the classification task. In addition to these, the two approaches proposed by Shibani et al. (2017) namely rule-based approach and machine learning approach using just (BOW) features were also added. The values are calculated by taking the average of the best performing classifiers in the four teamwork competency dimension from Shibani et al. (2017).

Table 4

Results of Different Approaches and Classifiers

		OVA	**KNN**	**DT**	**ET**	**ETs**	**RF**	**Rule-based**	**BOW**
Precision	Baseline*	0.81	0.76	0.76	0.73	0.82	0.84	0.68	0.84
	New^	0.84	0.79	0.78	0.74	0.84	0.85	-	-
Recall	Baseline	0.63	0.45	0.62	0.59	0.59	0.62	0.4	0.65
	New	0.68	0.48	0.68	0.64	0.64	0.66	-	-
F Score	Baseline	0.71	0.56	0.68	0.65	0.68	0.7	0.5	0.73
	New	0.74	0.58	0.72	0.68	0.72	0.74	-	-
Accuracy	Baseline	0.63	0.52	0.62	0.57	0.62	0.63	-	-
	New	0.67	0.53	0.66	0.61	0.65	0.67	-	-
Hamming Loss	Baseline	0.108	0.145	0.119	0.132	0.113	0.105	-	-
	New	0.097	0.139	0.109	0.126	0.103	0.097	-	-

Note. * Baseline dataset without new features. ^ Dataset with new features.

From the comparison of metrics in Table 4 based on the two sets of data, it can be seen that the performance of the models using the feature-engineered dataset was higher than the performance of models using the baseline data. The dataset with additional new features outperforms the baseline in terms of all the metrics. However, in order to look at a balanced classification model with an optimal balance for precision and recall, we focused on the F score. The F scores for all the classifiers were high for the dataset with new features. This clearly implies that the proposed linguistic features are useful to improve the classifier performance and demonstrates that feature engineering will be able to further improve the existing methods. A visual representation of the metrics is given in Figure 1.

Note that the results from Shibani et al. (2017) presented in Table 4 are an upper bound of performance (for BOW) that could be achieved rather than actual performance because they are taken from the average of the best performing classifiers in the four teamwork competency dimensions. This means that for example, the best performing classifier of recall under the COD category and the best performing classifier of recall under TES might not be the same type of classifier. Regardless of this, the comparison results show that the performance of classifiers using the proposed new linguistic features are able to outperform the best performing results from Shibani et al. (2017) for OVA and RT. In both cases, the BOW classifiers in Shibani et al. (2017) initially had better performance before feature engineering was incorporated.

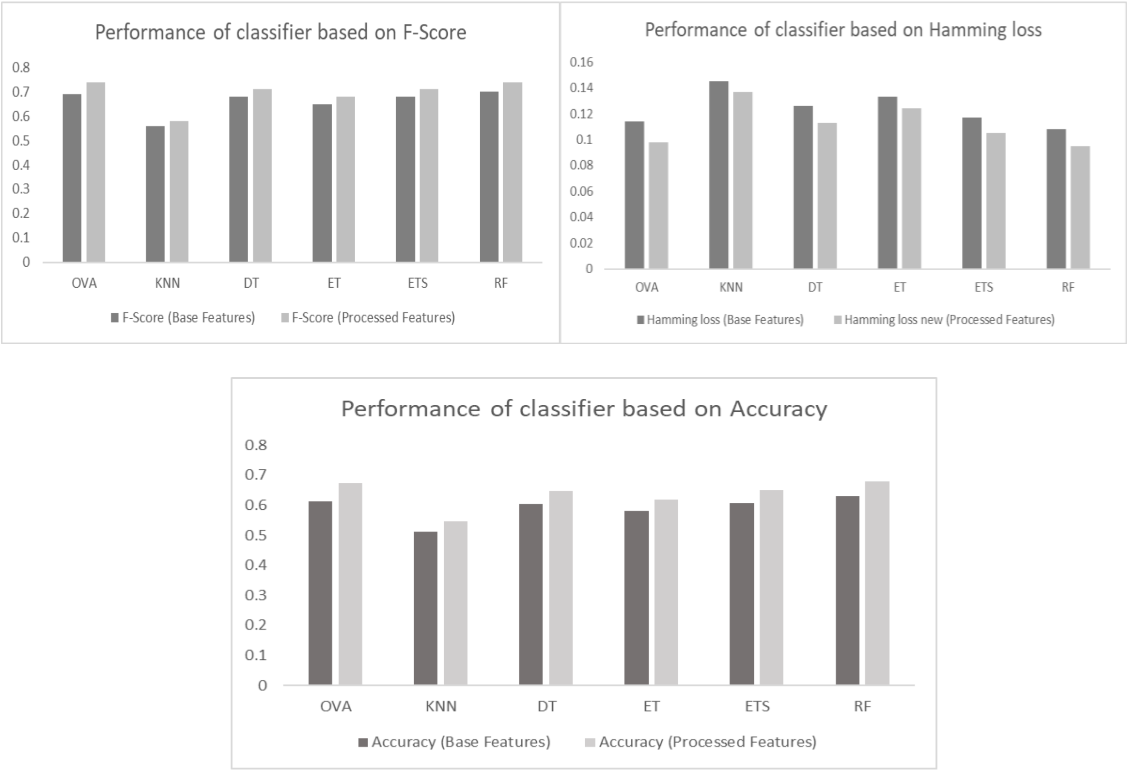

Figure 1. Performance metrics of all algorithms

The data coded by the system was compared with the human annotated data which is considered gold standard in order to check for reliability between the two sets of data. We calculated the reliability scores using the Krippendorf's alpha statistic in which reliability score was > 0.65 indicating that there is a good reliability between the human annotated and the system coded data.

With regard to the classification approaches, it can be seen from the F score measures that the adaptive algorithm approach and the problem transformation approach tie in performance. Further fine-tuning of parameters as well as inclusion of more features is required to decide which approach works best for our classification task. Finally, with regard to the classifiers used within the adaptive algorithm approach, the Random Forest classifier performed the best with an F Score of 0.74. At a dimension level, it can be seen from Table 2 that TES was easier to classify as individual F Score values for each of the classifier is relatively high when compared to the other dimensions. The classification performance for MPM was the weakest when compared to the other dimensions. This is mainly attributed to the difficulty in writing the rules for this dimension. More time and effort will be needed to address this need for refined rules that can code these dimensions effectively. Future work can focus on creating more features with better predictive capabilities and using a larger data set to train and test the models.

Table 5

F Score of the Different Classifiers at Dimension Level

	OVA	**KNN**	**DT**	**ET**	**ETS**	**RF**
COD	0.65	0.57	0.65	0.61	0.63	0.65
MPM	0.56	0.49	0.50	0.50	0.48	0.50
CCF	0.77	0.40	0.73	0.65	0.75	0.77
TES	0.87	0.79	0.87	0.85	0.86	0.88
F Score	0.71±0.13	0.56±0.16	0.68±0.15	0.65±0.14	0.68±0.16	0.70±0.16

4.1 Qualitative analysis

Identifying and creating strong predictive features can play a vital role in machine learning techniques. This section discusses on how the addition of predictive features helped the model in classifying text in a better way. Table 6 shows a few example chatlines that were coded correctly by the model. These chatlines are those that were predicted correctly mainly because of the addition of new features and were otherwise not actually tagged correctly while using the dataset without features.

For instance, a new feature to identify chatlines that denote progress of team members was created. This feature "F_PROG" will tag progress related chatlines by matching them with a list of words that denote progress such as "done", "accomplished", "finished", etc. Another example of a feature created would be the positive emotions feature (F_POSEMO) which tags chatlines that denote positive emotions like happy smileys and words such as "just joking", "thank you" etc. These examples briefed above explain the basis in which chatlines were tagged correctly.

Table 6

Examples of Chatlines that were Classified Correct

Sno	**Chatline without new features**	**Chatline with new features**	**Test Label**	**Predicted Label**
1	nothing already .. we are done	nothing already .. we are done F_PROG	COD	COD
2	stop please	stop please F_INS	MPM	MPM
3	okay den say use air purifier filler pos_emo	okay den say use air purifier filler pos_emo F_POSEMO F_ELABORATION	CSC TES	CSC TES

Table 7 shows example chatlines that were incorrectly predicted. A closer look at the data shows that the main reasons for the model to misclassify or not classify chatlines were irregularities in text and also the need for further refinement in the rules and the indicative terms dictionary that are used for these rules.

Table 7

Examples of Chatlines that were Classified Incorrect

Sno	**Chatline without new features**	**Chatline with new features**	**Test Label**	**Predicted Label**
1	talk about work	talk about work	COD	
2	^NAME^ copy and paste	^NAME^ copy and paste	COD MPM	COD
3	^NAME^ i suggest you go read the passage a few more timew	^NAME^ i suggest you go read the passage a few more timew F_INS	COD MPM	COD
4	air/smoke can travel anywhere	air/smoke can travel anywhere	CSC	
5	Trololololololtrolololololoololl	Trololololololtrolololololoololl	TES	
6	sure that seems like it works.	sure that seems like it works.	TES	

5. Conclusion and Future Work

This paper describes a text-mining approach to perform multi-label text classification of text-based chatlog into four teamwork competency dimensions. Unlike previous text classification approaches, this research has undergone an elaborate feature engineering process and produced a list of contextual features. A comparison study is then conducted to investigate the effect of incorporating the contextual features in various machine learning algorithms. The results show that by incorporating these features, it is possible to improve the classification scores regardless of the machine learning algorithm. Such a feature engineering exercise is promising and with more work, it is possible to further refine the features and achieve even better predictive capabilities. Future work will focus on further refining the system by training on larger datasets and also implementing complex rules. With regard to using larger datasets, since human coded data is considered gold standard, training using larger datasets also implies that more time and effort will be needed to manually code the datasets such that they can be used for training purposes. Once the model is reliably able to code all the four dimensions of teamwork, future data can be automatically coded using this system.

This current work as mentioned earlier will enable us to automatically identify and code the four dimensions of teamwork. This system can be seen as a way to automatically process the chat data after the student activity. The data coded will be aggregated by dimension to generate a micro-profile visualization of students' teamwork. For students, this visual analytic will serve as formative feedback and allow them to gain a better awareness of their teamwork dynamics and assist them in improving their teamwork. Teachers, on the other hand, can be empowered as this rapid assessment can complement existing observations and measures to provide timely and holistic feedback to students. Essentially, this approach serves as a basis to assess teamwork in student chat dialogue to enable students to gain a better awareness of their teamwork behaviors.

Acknowledgements

This paper refers to data and analysis from the research projects OER62/12EK and OER09/15 EK, funded by the Education Research Funding Programme, National Institute of Education, Nanyang Technological University, Singapore.

References

Aggarwal, C. C., & Zhai, C. (2012). *Mining Text Data*. doi:10.1007/978-1-4614-3223-4.

Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E. D., Gutierrez, J. B., & Kochut, K. (2017). A brief survey of text mining: Classification, clustering and extraction techniques. *arXiv preprint arXiv:1707.02919*.

Anders, A. (2016). Team communication platforms and emergent social collaboration practices. *International Journal of Business Communication*, 53(2), 224-261.

Anjewierden, A., Kolloffel, B., & Hulshof, C. (2007). Towards educational data mining: Using data mining methods for automated chat analysis to understand and support inquiry learning processes. In C. Romero, M. Pechenizikiy, T. Calders, & S. R. Viola (Eds.), *Applying data mining in e-learning* (pp. 23–32). *Proceedings of the International Workshop on Appling Data Mining in e-Learning (ADML'07)*. Crete, Greece, September 2007.

Chandrasekar, P., & Qian, K. (2016, June). The Impact of Data Preprocessing on the Performance of a Naive Bayes Classifier. In *Computer Software and Applications Conference (COMPSAC), 2016 IEEE 40th Annual* (Vol. 2, pp. 618-619). IEEE.

de Carvalho, A. C., & Freitas, A. A. (2009). A tutorial on multi-label classification techniques. In *Foundations of Computational Intelligence Volume 5* (pp. 177-195). Berlin, Heidelberg: Springer.

De Ferrari, L., & Mitchell, J. B. (2014). From sequence to enzyme mechanism using multi-label machine learning. *BMC bioinformatics*, *15*(1), 150.

Gibaja, E., & Ventura, S. (2015). A tutorial on multilabel learning. *ACM Computing Surveys (CSUR)*, *47*(3), 52.

Gopal, S., & Yang, Y. (2010, July). Multilabel classification with meta-level features. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval* (pp. 315-322). ACM.

Günal, S., Ergin, S., Gülmezoğlu, M. B., & Gerek, Ö. N. (2006, September). On feature extraction for spam e-mail detection. In *International Workshop on Multimedia Content Representation, Classification and Security* (pp. 635-642). Springer, Berlin, Heidelberg.

He, W. (2013). Examining students' online interaction in a live video streaming environment using data mining and text mining. *Computers in Human Behavior*, *29*(1), 90-102.

Hughes, R. L., & Jones, S. K. (2011). Developing and assessing college student teamwork skills. *New Directions for Institutional Research*, *2011*(149), 53-64.

Katakis, I., Tsoumakas, G., & Vlahavas, I. (2008, September). Multilabel text classification for automated tag suggestion. In *Proceedings of the ECML/PKDD* (Vol. 18).

Koh, E., Hong, H., & Tan, J. P. L. (2018). Formatively assessing teamwork in technology-enabled twenty-first century classrooms: exploratory findings of a teamwork awareness programme in Singapore. *Asia Pacific Journal of Education*, *38*(1), 129-144.

Read, J., Pfahringer, B., Holmes, G., & Frank, E. (2011). Classifier chains for multi-label classification. *Machine learning*, *85*(3), 333.

Rosa, K. D., & Ellen, J. (2009). Text classification methodologies applied to micro-text in military chat. In *Machine Learning and Applications, 2009. ICMLA'09. International Conference on* (pp. 710-714). IEEE.

Salas, E., Rosen, M. A., Burke, C. S., & Goodwin, G. F. (2009). The wisdom of collectives in organizations: An update of the teamwork competencies. In E. Salas, G. F. Goodwin & C. S. Burke (Eds.), *Team effectiveness in complex organizations. Cross-disciplinary perspectives and approaches* (pp. 39-79). New York: Routledge/Taylor & Francis Group.

Shibani, A., Koh, E., Lai, V., & Shim, K. J. (2017). Assessing the language of chat for teamwork dialogue. *Journal of Educational Technology & Society*, *20*(2), 224-237.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427-437.

Strijbos, J. W., & Sluijsmans, D. (2010). Unravelling peer assessment: Methodological, functional, and conceptual developments.

Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. *Information Processing & Management*, *50*(1), 104-112.