est similaire au travail entrepris, démarche qui peut être en lien avec la thématique
Ce papier date de 2019, démarche moderne



# Doc2Vec &Naïve Bayes: Learners' Cognitive Presence Assessment through Asynchronous Online Discussion TQ Transcripts

https://doi.org/10.3991/ijet.v14i08.9964

Hind Hayati (✉), Abdessamad Chanaa, Mohammed Khalidi Idrissi, Samir Bennani
Mohammed V University, Rabat, Morocco
hayati.hind@gmail.com

**Abstract**—Due to the lack of face to face interaction in online learning environment, this article aims essentially to give tutors the opportunity to understand and analyze learners' cognitive behavior. In this perspective, we propose an automatic system to assess learners' cognitive presence regarding their social interactions within asynchronous online discussions. Combining Natural Language Preprocessing, Doc2Vec document embedding method and machine learning techniques; we first make some transformations and pre-processing to the given transcripts, then we apply Doc2Vec method to represent each message as a vector that will be concatenated with LIWC and context features. The vectors are input data of Naïve Bayes algorithm; a machine learning method; that aims to classify transcripts according to cognitive presence categories.

**Keywords**—E-learning, asynchronous online discussion, Community of Inquiry, cognitive presence, text classification, doc2vec, machine learning, naïve Bayes, NLP, LIWC

## 1 Introduction

In the last decades, the development of communication technologies has brought important changes and improvement to the learning process. In fact, the new education relies on the use of information technologies to facilitate distant learning and information sharing regardless of time-space constraints. Asynchronous online learning, a new form of education, has become an important channel for knowledge construction and social interaction between learners and tutors[1] [2] [3]. One of the current challenges of today's online education is to predict students' behavior and engagement during the learning process in order to motivate them and help them succeed. However, the latent nature of engagement; especially the cognitive engagement which reflect learners' mental efforts with the learning environment; makes its prediction difficult and challenging[4].

Therefore, students' written messages in asynchronous online discussion contain relevant information about their critical thinking and mental efforts that should be captured and understood for sufficient and effective learning. In this regard, Community

of Inquiry (CoI); a successful validated model of distance education; provides a coding scheme instrument to analyze messages according to three essential dimensions; also known as presences; social, cognitive and teaching presence. In our work, we toggle the cognitive presence since it has a direct impact on learners' critical thinking and learning [5]. This later defines four phases that can learner achieve: Triggering Event, Exploration, Integration and Resolution. The CoI instrument describes each phase with a set of indicators of particular theorized socio-cognitive processes and takes each message as a unit of analysis. In fact, the coding scheme has been proven to be very useful and exhibits sufficient levels of inter-rater reliability to be considered as a valid construct [6].However, the use of this type of instrument is time-consuming and needs the intervention of experienced coders since it requires manual work. Manual analysis of textual data is considered painful and labor-intensive by many researchers due to the huge amount of information to sift through [7] [8] [9].

Different methods have been proposed to address this issue. Text Mining is the most used text-based method that can be efficient for analyzing textual data and capturing relevant and hidden patterns. TM is a process that combines different techniques as preprocessing, knowledge discovery and data mining[10]. In this perspective, we aim to automatically assess the level of cognitive presence in asynchronous online discussion transcripts using Text Mining approach. Specifically, we use NLP for processing natural language textual-data, then Doc2Vec, a document embedding method, for message representation before the classification step using Naïve Bayes as a machine learning algorithm for categorization.

The remainder of this article is organized as follows: in the first section, a literature review on automatic classification as well as the use of embedding techniques is presented. The second section presents the automatic assessment system for cognitive presence by defining the cognitive presence concept and describing the system architecture and its functioning. The implementation procedure and technical requirement will follow. Finally, we discuss results and draw a conclusion with perspectives of research.

## 2 Related Works

Recently, numerous researches and works are interested to use textual data and embedding methods for automatic classification in several fields.

The work in [11] proposes an automatic classifier for products presented by an integrated online-to-offline (O2O) service platform. In this perspective, they capture the semantics of words within the context of a product description by adapting doc2vec algorithm; a document embedding technique. In terms of classification accuracy, results have shown that doc2vec had significant improvement compared to bag-of-word modelling and word-level embedding.

Every text classification has a training data imbalance issue especially sentiment and emotion analysis where multiple categories produce skewed training data. Therefore, the contribution of[12] uses word embedding compositionality to develop an oversampling method. Results show the effectiveness of the method regarding the data

imbalance problem and the great improvement for both the binary sentiment classification and the multi-class emotion classification.

In the field of social media, text classification has an important role of understanding users' behavior. In this perspective, researches in[13] propose an adaptive approach namely TD2V; Twitter-based universal document representation; using doc2vec method. The method achieves a better classification accuracy regarding the existing ones.

Several researches used text classification for sentiment analysis, sometimes with TF-IDF technique or Bag-of-Word method but contributions in sentiment classification based on Doc2vec method have shown a high performance and accuracy than others[14] [15] [16].

According to the literature review, embedding techniques have shown their effectiveness in text-based classification system but still not been studied for classification problems in asynchronous online learning. This motivates us to examine the effectiveness of doc2vec method for assessing learners' cognitive presence level by analyzing their social interaction within asynchronous online discussion forums.

## 3 Automatic Assessment System for Cognitive Presence

In what follows the proposed automatic system for assessing cognitive presence and its functioning will be presented after an overview of cognitive presence.

### 3.1 Cognitive presence: A community of inquiry dimension

Community of Inquiry (CoI) is a pedagogical framework developed by Garrison, Anderson and Archer in 1999 [17]. Those later aims to develop students' critical thinking by the integration of inquiry-based learning throughout their lasting communication. Therefore, in the computer-mediated communication CoI represents a theoretical model based on social-constructivist pedagogies which describes three dimensions; cognitive presence, social presence and teaching presence; that work together to create a complete learning experience [18]. Ensuring the balance and the integration of these presences is essential for an effective learning.

Since it has a direct impact on student learning and critical thinking, cognitive presence represents the core construct in the CoI model. For Garrison, Anderson and Archer [5] "*Cognitive presence is defined as the extent to which learners are able to construct and confirm meaning through sustained discourse in a critical community of inquiry*". The cognitive presence construct is operationalized through the practical inquiry model; see Figure 1; that defines four phases of critical thinking process:

- ***Triggering event***—the phase in which a problem, issue or dilemma is defined.
- **Exploration**—when students are able to understand the basic nature of the problem and develop their critical reflection.
- ***Integration***—students can analyze and synthesize relevant information.

- ***Resolution***—new ideas and solutions are presented to the original problem. In our research, we focus on cognitive presence since it can capture learners' development of critical and deep thinking skills during the learning experience. In fact, we aim to assess learners' cognitive presence level according to their participation in asynchronous online discussion.

### 3.2 Contribution

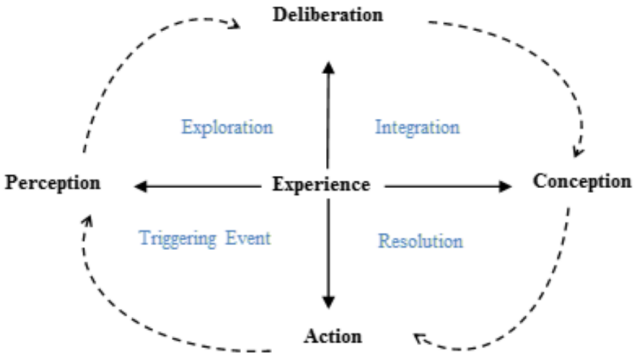


**Fig. 1.** Practical inquiry model for cognitive presence

We propose a system that can automatically assess learners' level of cognitive presence according to their social interaction within asynchronous online discussion forum using Text Mining and Doc2vec embedding method. Inspired by the word embedding technique, Doc2vec is used to extract surrounding word vectors that are specific to a document[19]. Generally, it's an unsupervised framework of learning continuous distributed vector representations for larger blocks of texts, such as sentences, paragraphs and documents. Therefore, we use Doc2vec as an embedding method to represent our messages into vectors so the machine-learning algorithm can understand them.

**Our objectives:** The proposed assessment is beneficial to both better understand learners' critical thinking and cognitive commitment as well as to develop a sophisticated learning environment for a better scaffolding. Among our objectives:

- Providing relevant information about the level of students' cognitive presence in a less time-consuming and labor-intensive way.
- Giving instructor the opportunity to use results for a better monitoring and ensure a continued functioning for an effective learning.

### 3.3 Architecture of the proposed system

To ensure our main objective; which is none other than automatically detect learner's level of cognitive presence; the proposed system has three main steps.

Since discussion forum transcripts are generally written in human natural language we first apply natural language preprocessing on the data set to have a clean and

understandable text. Then Doc2Vec; document embedding method; is used to represent messages as vectors concatenated with LIWC features and context discussion features. The final step consists of using a Naïve Bayes-based classifier in order to determine for each given message the CoI-cognitive presence phase it belongs to.

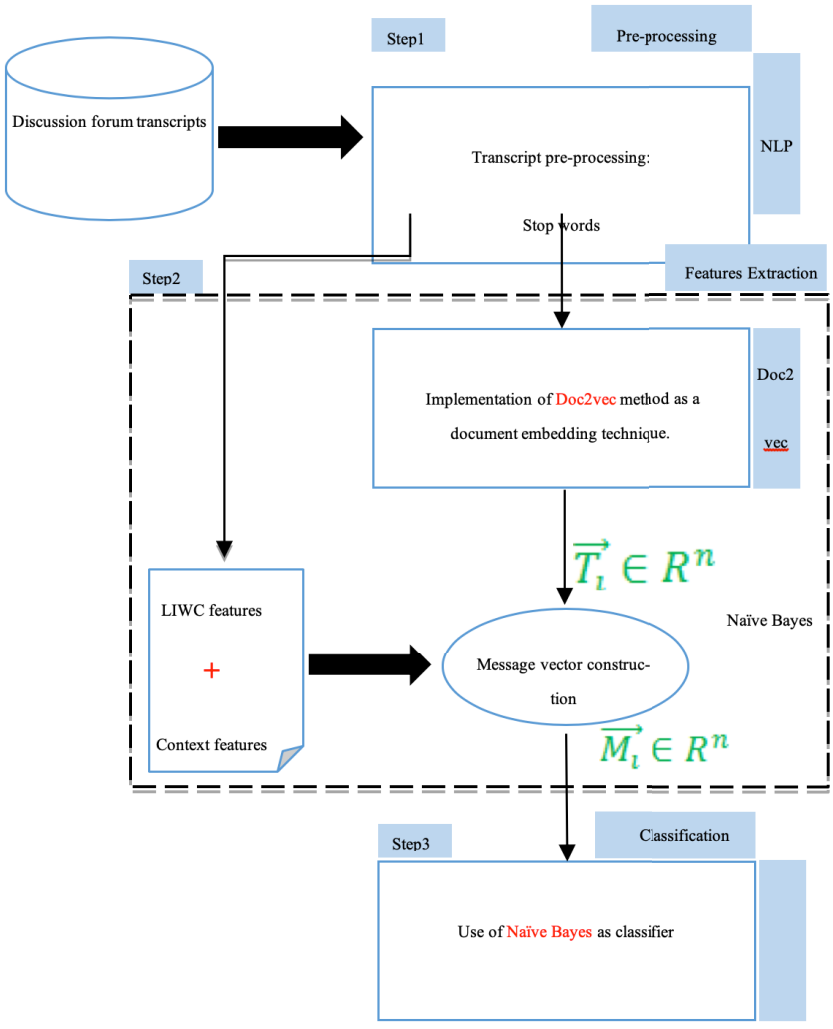


**Fig. 2.** Illustrate the architecture of the proposed system.

### 3.4 System Functioning

**Data set description:** The data set used in this study comes from discussion forum samples of different courses in software engineering offered through an online learning platform. The whole data set was coded by two experts according to the four levels of cognitive presence based on the coding scheme instrument defined by the Community

of Inquiry model. The inter-rater agreement was good: *percent agreement = 80%.*The disagreement was generally observed with exploration and integration phases due to the convergence of their meaning, at the end the two coders agree on the totality of messages codes and manage to find a middle ground.

**Pre-processing:** Since asynchronous online discussion generates a huge amount of transcripts in human natural language, we aim to do some preprocessing using NLP: Natural Language Preprocessing. This step relies on making the textual data clean and understandable by doing necessary transformations. For instance, we use three essential methods for processing text-based data:

---> remplacé par SpaCy

Corrspond à la classe preprocssing.py

- **Method 1**: Stop words: consists of removing all the irrelevant words that don't carry semantic information.
- **Method 2:** Spelling correction & Slang replacement: corrects any spelling mistakes, abbreviation and slang words like the word 'LOL' which can be corrected to the phrase 'Lot of Laugh'.
- **Method 3:** Stemming & Lemmatization: Stemming to find stem of a word by stripping prefix or suffix. Lemmatization to identify the lemma of a word. For example, the word assimilator is lemmatized to the word 'assimilate'

**Features extraction**

*Doc2vec***:** Text classification using machine learning techniques needs structured data at the input. However, our data set represents learners' transcripts through asynchronous online discussion which are unstructured data. At this level, we need to represent each message by a specific vector. Several representation methods exist and the most popular are:

- **Bag-of-Word (BOW):** A method that represents a message as a vector of term frequencies.

En cous de travail

- **Word2Vec**: A word embedding technique that predicts surrounding words of a given one. Word embedding composition is a method for superior data representation when data are skewed and sparse[12].
- **Doc2Vec**: Inspired by the word embedding techniques, it implements the method of distributed representation of paragraphs[15].

However, the BOW-based feature extraction ignores the semantic representation and words ordering. While Word2Vec has proven its effectiveness in a word-level representation. Therefore we adopt the Doc2Vec method since it can extract surrounding word vectors that are specific to a document, in our case the hall message, unit of analysis.

In our system and after implementing doc2vec method every message $i$ is represented by a vector $\vec{T_i}=\{t_{i,1},t_{i,2}, t_{i,3}, \ldots ,t_{i,n}\}$ where n is the vector_size value that represents the dimensionality of the feature vectors.

Besides the message representation, we extract two types of indicators; LIWC (Linguistic Inquiry and Word Count) features which are indicative of different psychological processes including cognitive one and discussion context features that represent relevant information about each message.

*LIWC features*: In our approach, we try to understand how cognitive presence can be captured within discourse. So, we use LIWC that represent a transparent program for text analysis to make a count of words in psychologically meaningful categories[20]. LIWC program analyzes written messages and the text analysis module compares each word detected in the text against adictionary that identifies which words are associated with which psychologically-relevant categories (e.g. cognitive, affective, social, etc.). In our case, we analyze according to the cognitive category. Thus, after the program has read and accounted for all words in the given message, it gives the results as percentage of total words that match the cognitive categorys

*Discussion context features*: For more precision and credible results, we incorporate context information in our feature space. Therefore, we include some features related to the transcripts like:

Peut être inclus après mise en place des classes utterances, dialogues

- Message width: an integer variable indicating the number of words within a message.
- Message depth: an integer variable showing the position of a given message within a discussion.
- Number of children belonging to a message: an integer variable indicating the number of replies a specific message received.
- Number of votes: an integer variable showing the number of votes collected for a given message.

After the step of feature extraction, we concatenate the obtained values of LIWC and discussion context features with the vector $\overrightarrow{T_l}$ in order to have a more specific vector $\overrightarrow{M_l}$ for each message.

Algorithme de -------> classification pas encore choisi

sklearn pour faire des tests

**Classification:** Within text classification process the application of machine learning algorithm is essential. In our system, we choose to use Naïve Bays algorithm to classify learners' messages according to four categories of cognitive presence. Naïve Bayes represents a classical method for document classification [21] [22] [23] based on Bayes' theorem

Considered as a supervised learning method NB can predict class membership posterior probabilities. For instance, the probability that a given message belongs to a particular cognitive presence category.

In our work, we attempt to determine in which class a given messageis belonging to knowing that we have C classes (e.g. C=4; the four phases of the CoI-Cognitive presence). Therefore, NB-classifier will predict that$\overrightarrow{M_l}$belongs to the class $C_j$ having the highest posterior probability, conditionedon $\overrightarrow{M_l}$.

Let's take the message number $\ell$. $\overrightarrow{M_l}$ is predicted to belong to the class $C_i$ if and only if:

$P(C_i|\overrightarrow{M_l}) > P(C_j|\overrightarrow{M_l})$ for $1 \leq j \leq k, j \neq i$, k: number of classes

Thus we detect the class that maximizes $P(Ci|\overrightarrow{M_l})$. In fact, Ciis called the maximum posterior hypothesis. By Bayes' theorem:

$$P(C_i|\overrightarrow{M_l}) = \frac{P(\overrightarrow{M_l}|C_i)P(C_i)}{P(\overrightarrow{M_l})}$$

### 3.5 Implementation

In our case, we use python programming language to implement our classifier with several software packages:

- For NLPwe used Natural Language Toolkit NLTK 3.3 for preprocessing[24].
- For implementing doc2vec method we used gensim's doc2vec library and numpy package for multidimensional array object.
- For LIWC features we used an online API[25]
- For developing Naïve Bayes classifier, we used Bernoulli NBscikit-learn package withscipy math library[26].

## 4 Results

The purpose of the implementation is training the proposed system and evaluate its efficiency and accuracy. Thus, we start by training the doc2vec model with training data, four documents which are tagged by the four phases of cognitive presence (TE, EX, IN and RE), and each document includes messages belonging to a specific phase Figure 3. Then the model will generate vectors for each document and predict vectors for testing dataset using inference function in doc2vec. The objective being to have a document vector for every new document.

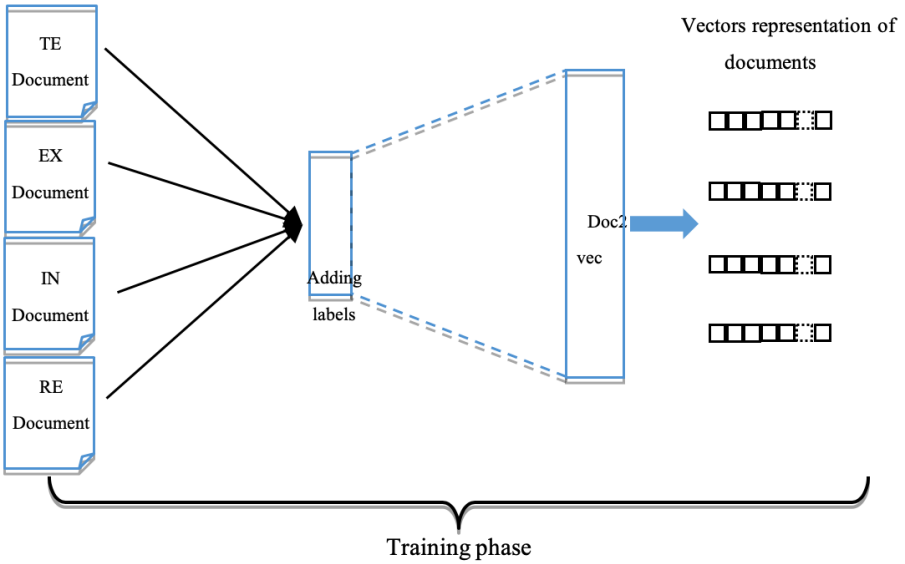


**Fig. 3.** Training Doc2vec model

After that, we start training Naïve Bayes algorithm with messages' vectors and to be sure of its sufficiency in our case regarding other algorithms, we compare in table 1.

Accuracy results (*classification accuracy, cohen's K, recall precision and f1 score*) for three algorithms: Naïve Bayes, SVM, Logistic Regression

**Table 1.** Accuracy measures for machine learning algorithm

| | **Naïve Bayes** | **SVM** | **Logistic Regression** |
|---|---|---|---|
| Classification Accuracy | 0.7647 | 0.7058 | 0.6470 |
| Cohen's K | 0.6866 | 0.6064 | 0.5299 |
| Recall precision | 0.689 | 0.689 | 0.753 |
| F1 score | 0.7705 | 0.6784 | 0.6605 |

We can see that NB have shown the best results for classification accuracy, cohen's k and F1 score. But for the recall precision LR wins with a difference of 0.07 which make NB the adequate algorithm in our case. Figure 3. Depicts the precision-recall measure for each class, in our case cognitive presence phases (TE, EX, IN, RE)

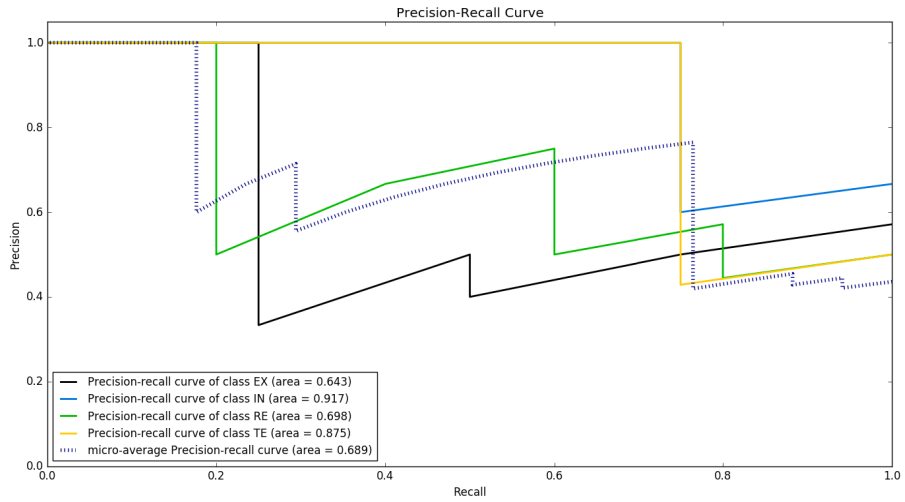


**Fig. 4.** Precision-Recall Curve

We remark that Integration phase achieves the best recall score followed by the Triggering Event phase and Resolution in the third place and finally the exploration phase. The differences are usually due to the correlation between phases and confusion that could coders have between levels.

## 5 Conclusion

In this paper, we aim to develop an automatic system based on text classification for assessing learners' cognitive presence by coding their discussion messages according to its four levels. The obtained classification accuracy is ~ 0.77 and Cohen's $\kappa \sim 0.69$. Therefore, the performance of the proposed system is considered to be in the range of

a substantial level of agreement. Our approach is essentially based on learners' social-interaction in asynchronous online discussion to analyze and assess their critical thinking and cognitive behavior. In this perspective, we start by collecting students' transcripts and making some transformation using NLP methods in order to prepare data for the step of features extraction. This later combine three types of features:

- ***Doc2vec feature vectors***: since we have text-based data and machine learning don't accept unstructured data, we use doc2vec embedding method to represent each message by a specific vector.
- ***LIWC feature***: percentage of total words that match the cognitive category in a given message.
- ***Discussion context feature***: context features related to the transcripts like: Message width, Message depth, Number of children belonging to a message, Number of votes.

At the last phase of the proposed system, the resulted vectors are inputs of a Naïve Bays classifier that categorize messages into the four levels of cognitive presence.

As perspective, we intend to use results for predicting learners' cognitive engagement and give tutors a real time interaction with learners for a better monitoring.

## 6 References

[1] X. Wang, D. Yang, M. Wen, K. Koedinger, and C. P. Rosé, "Investigating how student ' s cognitive behavior in MOOC discussion forums affect learning gains," *Proc. 8th Int. Conf. Educ. Data Min.*, pp. 226–233, 2015.

[2] D. Wu and S. R. Hiltz, "Predicting learning from asynchronous online discussions," *J. Asynchronous Learn. Netw.*, vol. 8, no. 2, pp. 139–152, 2004.

[3] H. Hayati, M. Khalidi Idrissi, and S. Bennani, "Applying text mining to predict learners' cognitive engagement," in *ACM International Conference Proceeding Series*, 2017.

[4] V. Aleven, "Rule-Based Cognitive Modeling for Intelligent Tutoring Systems," Springer, Berlin, Heidelberg, 2010, pp. 33–62. https://doi.org/10.1007/978-3-642-14363-2_3

[5] D. R. Garrison, T. Anderson, and W. Archer, "Critical thinking, cognitive presence, and computer conferencing in distance education," *Am. J. Distance Educ.*, vol. 15, no. 1, pp. 7–23, 2001. https://doi.org/10.1080/08923640109527071

[6] V. Kovanović *et al.*, "Towards automated content analysis of discussion transcripts," *Proc. Sixth Int. Conf. Learn. Anal. Knowl. - LAK '16*, pp. 15–24, 2016. https://doi.org/10.1145/2883851.2883950

[7] A. Darabi, M. C. Arrastia, D. W. Nelson, T. Cornille, and X. Liang, "Cognitive presence in asynchronous online learning: A comparison of four discussion strategies," *J. Comput. Assist. Learn.*, vol. 27, no. 3, pp. 216–227, 2011. https://doi.org/10.1111/j.1365-2729.2010.00392.x

[8] V. Kovanović, D. Gašević, and G. Siemens, "A Novel Model of Cognitive Presence Assessment Using Automated Learning Analytics Methods About Analytics for Learning (A4L)," no. January, 2017.

[9] J. J. Mills, "A Mixed Methods Approach To Investigating Cognitive Load And Cognitive Presence In An Online And Face-To-Face College Algebra Course," 2016.

[10] A. C. M. Fong, S. C. Hui, and G. Jha, "Data mining for decision support," *IT Prof.*, vol. 4, no. 2, pp. 9–17, 2002. https://doi.org/10.1109/MITP.2002.1000455
[11] H. Lee and Y. Yoon, "Engineering doc2vec for automatic classification of product descriptions on O2O applications," *Electron. Commer. Res.*, pp. 1–24, 2017.
[12] R. Xu, T. Chen, Y. Xia, Q. Lu, B. Liu, and X. Wang, "Word Embedding Composition for Data Imbalances in Sentiment and Emotion Classification," *Cognit. Comput.*, vol. 7, no. 2, pp. 226–240, 2015. https://doi.org/10.1007/s12559-015-9319-y
[13] L. Q. Trieu, H. Q. Tran, and M.-T. Tran, "News Classification from Social Media Using Twitter-based Doc2Vec Model and Automatic Query Expansion," *Proc. Eighth Int. Symp. Inf. Commun. Technol. - SoICT 2017*, pp. 460–467, 2017. https://doi.org/10.1145/3155133.3155206
[14] y6y6S. Lee, X. Jin, and W. Kim, "Sentiment classification for unlabeled dataset using Doc2Vec with JST," in *Proceedings of the 18th Annual International Conference on Electronic Commerce e-Commerce in Smart connected World - ICEC '16*, 2016, pp. 1–5. https://doi.org/10.1145/2971603.2971631
[15] Q. V. Le and T. Mikolov, "Distributed Representations of Sentences and Documents," vol. 32, 2014.
[16] F. Colace, M. de Santo, and L. Greco, "Safe: A sentiment analysis framework for e-learning," *Int. J. Emerg. Technol. Learn.*, vol. 9, no. 6, pp. 37–41, 2014. https://doi.org/10.3991/ijet.v9i6.4110
[17] D. R. Garrison, T. Anderson, and W. Archer, "Critical Inquiry in a Text-Based Environment: Computer Conferencing in Higher Education," *Internet High. Educ.*, vol. 2, no. 2–3, pp. 87–105, 1999. https://doi.org/10.1016/S1096-7516(00)00016-6
[18] H. Hind, M. KHALIDI IDRISSI, and S. BENNANI, "Automatic Assessment of CoI-Cognitive Presence within Asynchronous Online Learning," in *2018 17th International Conference on Information Technology Based Higher Education and Training (ITHET)*, 2018, pp. 1–5.
[19] D. Kim, D. Seo, S. Cho, and P. Kang, "Multi-co-training for document classification using various document representations: TF–IDF, LDA, and Doc2Vec," *Inf. Sci. (Ny).*, vol. 477, pp. 15–29, Mar. 2019. https://doi.org/10.1016/j.ins.2018.10.006
[20] Y. R. Tausczik and J. W. Pennebaker, "The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods," J. Lang. Soc. Psychol., vol. 29, no. 1, pp. 24–54, Mar. 2010. https://doi.org/10.1177/0261927X09351676
[21] I. Rish and I. Rish, "An empirical study of the naive bayes classifier," 2001.
[22] K. M. Leung, "Naive bayesian classifier," *Polytech. Univ. Dep. Comput. Sci. Risk Eng.*, 2007.
[23] M. Pazzani and D. Billsus, "Learning and Revising User Profiles: The Identification of Interesting Web Sites," *Mach. Learn.*, vol. 27, no. 3, pp. 313–331, 1997. https://doi.org/10.1023/A:1007369909943
[24] "Natural Language Toolkit — NLTK 3.3 documentation." [Online]. Available: https://www.nltk.org/. [Accessed: 07-Nov-2018].
[25] "LIWC: Linguistic Inquiry and Word Count." [Online]. Available: http://www.liwc.net/tryonlineresults.php. [Accessed: 18-Oct-2018].
[26] "Scikit-learn: machine learning in Python — scikit-learn 0.20.0 documentation." [Online]. Available: https://scikit-learn.org/stable/. [Accessed: 07-Nov-2018].

## 7 Authors

**Hind Hayati** is a Ph.D. Candidate at the Laboratory of Research in Computer Science and Education, Mohammadia School of Engineers, Mohammed V University of Rabat, Morocco. My research project is about the use of educational technology and community of inquiry to improve Students' Learning and cognitive behavior. My interests lie in the fields of Text mining and machine learning techniques, Educational engineering, Community of inquiry and cognitive engagement. Email: hayati.hind@gmail.com

**Abdessamad Chanaa** is a Ph.D. Candidate at the Laboratory of Research in Computer Science and Education, Mohammadia School of Engineers, Mohammed V University of Rabat, Morocco. My research project is about the use of contextual modelling recommendation in Massive Open Online Courses (MOOCs), by creating a personalized profile of each learner using machine learning and deep learning techniques. Email: abdessamad.chanaa@gmail.com

**Mohammed Khalidi Idrissi** is a Full Professor at Mohammadia School of Engineers. Doctorate degree in Computer Science in 1986, PhD in Computer Science in 2009; Former Assistant chief of the Computer Science Department at the Mohammadia School of Engineers (EMI); Pedagogical Tutor of the Computer Science areas at the Mohammadia School of Engineers (EMI) Professor at the Computer Science Department-EMI; 34 recent publications papers between 2014 and 2017; Ongoing research interests: Software Engineering, Information System, Modeling, MDA, ontology, SOA, Web services, eLearning content engineering, tutoring, assessment and tracking. Email: khalidi@emi.ac.ma

**Samir Bennani** is a Full Professor and Deputy Director of students and academic affairs at Mohammadia School of Engineers. Engineer degree in Computer Science in 1982; PhD in Computer Science in 2005; Professor at the Computer Science Department-EMI; 34 recent publications papers between 2014 and 2017; Ongoing research interests: SI, Modeling in Software Engineering, Information System, eLearning content engineering, tutoring, assessment and tracking. Email: sbennani@emi.ac.ma

Article submitted 2018-12-05. Resubmitted 2019-02-18. Final acceptance 2019-03-03. Final version published as submitted by the authors.