Assessing Student-Generated Design Justifications in Engineering Virtual Internships

Vasile Rus, Dipesh Gautam,
Department of Computer Science
Institute for Intelligent Systems
The University of Memphis
Memphis, TN 38152
{vrus,dgautam}@memphis.edu

Zachari Swiecki, David W. Shaffer
Department of Educational Psychology
University of Wisconsin-Madison
Madison, WI 53706
{swiecki,dws}@wisc.edu

Arthur C. Graesser
Department of Psychology
Institute for Intelligent Systems
The University of Memphis
Memphis, TN 38152
graesser@memphis.edu

ABSTRACT

Engineering virtual internships are simulations where students role play as interns at fictional companies, working to create engineering designs. To improve the scalability of these virtual internships, a reliable automated assessment system for tasks submitted by students is necessary. Therefore, we propose a machine learning approach to automatically assess student generated textual design justifications in two engineering virtual internships, *Nephrotex* and *RescuShell*. To this end, we compared two major categories of models: domain expert-driven vs. general text analysis models. The models were coupled with machine learning algorithms and evaluated using 10-fold cross validation. We found no quantitative differences among the two major categories of models, domain expert-driven vs. general text analysis, although there are major qualitative differences as discussed in the paper.

Keywords

Virtual internships, machine learning, auto-assessment, epistemic frame theory

1. INTRODUCTION

In virtual internships, students play the role of interns in a virtual training environment. In engineering virtual internships, such as *Nephrotex* (NTX) and *RescueShell* (RS), students research and create multiple engineering designs [1]. As part of their design process, they regularly submit written work in the form of electronic engineering notebooks that are assessed by human judges. This human assessment is labor intensive, time consuming, and error-prone under certain circumstances such as time pressure. Furthermore, prior work has suggested that the reliability of human assessments can vary depending on the traits of the assessor, their experience, and the types of problems being assessed [14]. Thus, an automated assessment method that could provide efficiency in terms of time and cost as well as improved reliability is much needed. Our work presented here constitutes a step in this direction.

In the present study, we explored various models for automatically assessing notebooks in the engineering virtual internships NTX and RS. The content of these notebooks varies; however, in this study we focus on only one type of notebook in which students must justify their engineering designs by typing a short, free-text justification.

We have experimented with models that emulate an expert analysis of the student notebook entries as well as models derived from general textual analysis features. It should be noted that our work differs from previous attempts which rely on a semantic similarity approach, i.e. measuring how semantically close a student-generated response is to an ideal, expert-generated response as in [6].

The domain expert-driven models incorporate theoretically driven, content-based features identified by human experts such as “referencing any performance parameter such as cost”, which is a general design feature because it applies to all engineering designs in NTX and RS, or “indicating the power source”, a feature specific to the concrete task of designing an exoskeleton, which was the focus of the RS internship and not NTX. A challenge with the domain expert-driven models is that the features are specific to either the type of task, e.g. engineering design, or the concrete task itself, e.g. design an exoskeleton. This results in a scalability issue as these models must be redesigned manually by domain experts when moving to a new domain, new type of task, and/or a new concrete task. However, the net theoretical advantage of these domain expert-driven models is that they are tailored to the task at hand and therefore are expected to yield very good performance. These models also afford the ability to create automatic and tailored feedback to students given their task-specific diagnostic capabilities.

The other category of models that we used rely on general text analysis features inspired from previous work on automated essay scoring [2,5,13] and text analysis software tools such as Coh-Metrix [4] and LIWC [7]. For instance, in automated essay scoring the length in words of the essay, i.e. the number of all word occurrences or word tokens, is by far the best predictor of essay quality. Coh-Metrix is a software package that calculates the coherence of texts in terms of co-reference, temporal cohesion, spatial cohesion, structural cohesion, and causal/intentional cohesion. LIWC (Linguistic Inquiry and Word Count) uses a word count strategy to characterize texts along a number of dimensions that include standard language categories (e.g., articles, prepositions, pronouns), psychological processes (e.g., positive and negative emotion word categories), and traditional content dimensions (e.g., sex, death, home, occupation).

The key advantage of the general text analysis models is that they are generally applicable across types of tasks, specific tasks, and domains. In addition, the general text analysis features are relatively cost-effective and easy to derive from the data compared to features derived by domain experts, which require (significantly more) human time and effort.

In this paper, we explore the predictive power of the two major categories of models mentioned above, domain-expert vs. general text analysis, in conjunction with a number of machine learning algorithms such as decision trees, naïve Bayes, Bayes Nets, and logistic regression. Furthermore, we employed an ensemble of classifiers approach in order to boost the performance of individual models. We conclude the paper with a qualitative assessment of the relative benefits of the proposed models for virtual internships by considering their predictive value, the labor involved in their development, and their ability to provide interpretable assessments for students.

2. BACKGROUND

We review in this section prior work on assessing students' open-ended responses with an emphasis on prior work in the area of educational technologies.

Automated essay scoring systems [2,5,13] have been developed for more than two decades as a way to tackle the costs, reliability, generality, and scalability challenges associated with assessing student generated open-ended responses to essay prompts. There are a number of systems available for automated essay scoring, some of which are commercial. It is beyond the scope of this paper to offer a thorough review of the work in this area. We limit ourselves to noting that the focus on automated essay scoring is on the argumentative power of an entire essay while in our case the focus is on required (design) items that must be present in paragraph-like justifications. This entails that style and higher-level constructs such as rhetorical structure are less important in our task as opposed to the essay scoring task and that factors that focus more on content measures are highly important. Given these differences and the fact that the two most predictive factors of essay quality are also content related, we included in our models the following two features: word count, i.e. total number of word occurrences or tokens in student justifications, and content word count, i.e. the total number of content word occurrences (nouns, verbs, adjectives, and adverbs).

Directly relevant to our study is previous work by Rus, Feng, Brandon, Crossley, and McNamara [8] who studied the problem of assessing student-generated paraphrases in the context of a writing strategy training tutoring system. One of the strategies in this tutoring system is paraphrasing. As the system is supposed to prompt students to paraphrase and then provide feedback on their paraphrases, Rus and colleagues collected a large corpus of student-generated paraphrases and analyzed them along several dozen linguistic dimensions ranging from cohesion to lexical diversity obtained from Coh-Metrix [4]. There are significant differences between their work and ours. First, we deal with justifications which can vary in length from a few words to a full paragraph as opposed to explicitly elicited paraphrases of target sentences. Second, we do use extra features to build our models besides the Coh-Metrix indices. Third, we assess the student generated justifications as acceptable or unacceptable (i.e., correct or incorrect). We could eventually investigate finer levels of correctness, e.g. on a scale from 1-5, which we plan to do as part of our future work.

Williams and D'Mello [15] worked on predicting the quality of student answers (as error-ridden, vague, partially-correct or correct) to human tutor questions, based on dictionary-based dialogue features previously shown to be good detectors of cognitive processes (cf. [15]). To extract these features, they used LIWC (Linguistic Inquiry and Word Count; [6]), a text analysis software program that calculates the degree to which people use various categories of words across a wide array of texts genres. They reported that pronouns (e.g. I, they, those) and discrepant terms (e.g. should, could, would) are good predictors of the conceptual quality of student responses. Like Williams and D'Mello, we do use LIWC to analyze student notebooks' justifications. Furthermore, we employ expert-identified features and features from Coh-Metrix and automated essay scoring.

Prior work by Rus, Lintean, and Azevedo [9] investigated the performance of several automated models designed to infer the mental models of students participating in an intelligent tutoring system (ITS). The ITS was designed to teach students self-regulatory processes while they were learning about science topics such as the human circulatory system. Rus and colleagues used two methods, a content-based method and a word-weighting method, to derive features for their models. While our present work does not investigate models using word-weighting methods, we do investigate models using content-based features.

The content-based features used by Rus and colleagues included a taxonomy of relevant biology concepts derived by human experts, expert annotated pages of content from the ITS, and expert-generated paragraphs. In the present study, the content-based features, or domain-expert (DE) features, we used consist of discourse codes developed by human experts. Discourse codes indicate the presence or absence of specific concepts in student talk, or in this case, student written work. The DE features were developed through a grounded analysis of student design justifications collected from engineering virtual internships [3].

The learning that occurs in engineering virtual internships can be characterized by epistemic frame theory. This theory claims that professionals develop epistemic frames, or the network of skills, knowledge, identity, values, and epistemology that are unique to that profession [11]. For example, engineers share ways of understanding and doing (knowledge and skills); beliefs about which problems are worth investigating (values), characteristics that define them as members of the profession (identity), and a ways of justifying decisions (epistemology). In this study, we used epistemic frame theory to guide the development of the DE features. In prior work, elements of the engineering epistemic frame have been operationalized as discourse codes and used to assess engineering thinking in virtual internships [1]. In this study, the DE features we identified correspond to elements of the engineering epistemic frame that relate to justifying design decisions. The presence or absence of these features in a student's written work thus represents elements of the engineering epistemic frame that are present or lacking.

In sum, we used some of the features described by the above researchers in our work, such as word count, as well as novel features, e.g. features based on the engineering epistemic frame.

3. ENGINEERING VIRTUAL INTERNSHIPS

In this study, we examined student written work collected from the engineering virtual internships, *Nephrotex* (NTX) and *RescueShell* (RS). In NTX, students work in teams to design filtration membranes for hemodialysis machines, while in RS, student teams design the legs of a mechanical exoskeleton used by rescue workers.

All interactions in virtual internships take place via a website in which students communicate with their teams using email and chat. During the internships, students research and create engineering designs in two cycles. In each cycle, students design five prototypes and later receive performance results for each prototype which they have to analyze and interpret.

During their design process, students submit records of their work via electronic notebook entries for each substantive task they complete, including summarizing research reports and justifying design decisions. The expectations of notebook entries are outlined in prompts, which students receive via email in the virtual internship website. Each notebook that students submit is divided into notebook sections, i.e., separate text fields for items that are defined by the email prompts. In this study, we analyzed notebook sections in which students provided justifications for their prototype design decisions.

Once students complete each notebook section, they submit the notebooks to trained human raters for assessment. In the fiction of the virtual internships, these raters play the role of more senior employees in the company who act as *mentors* to the students. The role of the mentors is to answer student questions and lead team discussions, in addition to assessing student work.

Once a mentor receives a notebook, they assess each section as acceptable or unacceptable using provided rubrics. The assessment system used by the mentors automatically generates pre-scripted feedback corresponding to the assessment given to each section. Currently, this feedback is generic in the sense that it does not respond to the particulars of a student's response. For example, an assessment of unacceptable on a notebook section requiring a summary generates feedback that (1) informs the student that the section was unacceptable, (2) reminds them of the content they were asked to summarize, and (3) points them to the documents they were asked to summarize. This automated feedback does not inform the student exactly why the section was rated as unacceptable. However, the mentor does have the option to compose specific feedback for the student if they wish.

Our work here moves us towards a more automated and student-tailored assessment and feedback mechanisms which could have significant impact on the economy of scaling virtual internships to all students, anytime, anywhere via Internet-connected devices.

4. EXPERIMENTS AND RESULTS

We describe first the data set we used in our experiments before presenting the experiments and results obtained with the models.

4.1 Data Set

In this study, we analyzed notebook sections from the NTX and RS virtual internships in which students justified their engineering design decisions. In these notebook sections, students were required to include the design input choices they selected—that is, their design specifications, and a justification explaining why this design was chosen for testing.

Mentors assessed these notebook entries as acceptable or unacceptable in real-time during the virtual internship using the following rubric:

1. Listed their design specifications
2. Included a justification referencing at least one design specification.

Acceptable justification may include:

1. Prioritizing attributes
2. Referencing internal consultant requests
3. The performance of a design specification on a specific attribute
4. Experimental justifications (e.g., holding design specifications constant)

To select data for this study, we randomly sampled 298 justification sections from 20 virtual internship sites, i.e. datasets corresponding to 20 schools where the virtual internships were implemented. Twelve were NTX sites and eight were RS sites. Of the 298 justifications sampled, 146 were from NTX and 152 were from RS. Students were given the same prompts for justification sections in NTX and RS. In addition, the same rubrics were used by raters in NTX and RS. Thus, we combined data from RS and NTX to train our models.

As described above, justification sections were originally assessed by mentors during the virtual internship in real time. The mentors were trained to assess notebook section, but they were not experts in the domain of engineering or the content of the virtual internships. In addition, they had to assess notebook sections under time constraints and while completing their other responsibilities as a mentor. For example, they could have to respond to student questions via chat while assessing. Thus, to obtain potentially more valid and reliable assessments for model training, the justification sections in this study were re-assessed by more experienced raters that did not face the constraints placed on the mentors. We found that the agreement between the human mentors and our experienced raters on the 298 student justifications we used in this work was kappa = 0.271. This value is very low, indicating that mentors' assessments are not reliable, as we suspected.

Each justification section was re-assessed by two new raters, benchmark rater 1 (BE1) and benchmark rater 2 (BE2). BE1 had over two years of experience rating notebook sections from virtual internships and had contributed to the content development of both NTX and RS. BE1 was thus considered an expert rater for the purposes of this study. BE2 was a less experienced rater trained to assess justification sections. BE1 and BE2 assessed all 298 justification sections using the rubric above and agreed on one final judgement (acceptable or unacceptable) for each justification. Their inter-annotator reliability as measured by kappa was 0.767**.** Table 1 includes examples of notebook sections from NTX assessed as acceptable and unacceptable by the benchmark raters. About 73% of the instances in the data set were rated positively by the BEs. The distribution of positive and negative instances is shown in Table 2.

4.2 Feature Selection

As already mentioned, we focused on two major categories of models: models that rely on domain-experts (DE) versus models that rely on more general textual analysis features. We developed the DE features through a grounded analysis [3] of a sample of 98 justification sections. These features were developed by two researchers who re-assessed the sample and developed discourse codes corresponding to what they attended to while assessing. Next, we automated these codes using the *nCoder*, a tool for developing and validating automated discourse codes that relies on authoring targeted regular expressions for each of the expert-identified codes [12]. These codes were included as features in our models (see Table 3 for descriptions).

Table 1. Example of Acceptable and unacceptable notebooks from the virtual internship *Nephrotex*

Notebook entry	Assessment
Design Specifications: PAM, Vapor, Negative Charge, 4 % Justification: This prototype was altered slightly from the original with this material by changing from 2% CNT to 4%. This is an attempt to increase reliability without hindering flux or blood cell reactivity.	**Acceptable**
Design Specifications: PAM, Vapor, Negative Charge, 2.0 Justification: These specificaions ran best for PAM material	**Unacceptable**

Table 2. Distribution of human-ratings in the 298 instances.

Human Rating	#Instances
Acceptable	217
Unacceptable	81
Total	298

The general textual analysis features were further divided by their source into the following three categories: features inspired from automated essay scoring (ES) research, features obtained with the automated tool for textual analysis Coh-Metrix, and features obtained with the automated tool for textual analysis LIWC. This categorization of the general textual analysis features is needed for several reasons. First, the various sources capture different aspects of a text. Second, this categorization allows us to conduct ablation studies in which we assess the contribution of each major category of features to solving the task at hand. It should be noted that there is overlap among the features from various groups/sources. For instance, the WC (LIWC), DESWC (Coh-Metrix), and Word_Count (DE) features are all counts of white-spaces in a target text, i.e. justifications in our case. These features are slightly different from the token Count feature in the ES group which counts number of tokens after applying the Stanford tokenizer tool. Similar features will not end up in the same models if they correlate highly, as explained next.

Not all features have equal predictive power and having redundant or irrelevant features can decrease the performance of the models. Therefore, we had a feature selection step keeping features that have low correlation with each other (<.70). When two features in a model had a correlation greater than .70 of them was dropped. For instance, from the LIWC and Coh-Metrix groups of features the features selected via this process were: WC, SIXLTR, adverbs, verbs, DESSC, DESSL, DESSLd, PCNARz, PCCONNp (See Table 3 for descriptions). The feature selection step was needed given that we worked with various machine learning algorithms, some of which do not have a feature selection process linked to them, e.g. the stepwise variable selection in some regression implementations.

4.3 Results

We experimented with the proposed models in conjunction with a number of classification algorithms including decision trees, naïve Bayes, Bayes Nets, and logistic regression. We present here the results obtained with the logistic regression classifier as it yielded the best results overall. The models were validated using 10-fold cross validation. Performance was measured using standard measures such as accuracy, false positive rate, precision, recall, F-measure, and kappa statistic. The false positive rate, the percentage of true negatives predicted as positives, is of special interest because it gives us an idea of how many justifications are deemed correct when in fact are not, by a particular method. That is, it indicates how many opportunities for feedback a specific method might miss as a justification deemed correct means there is no need for specific feedback to improve it. The evaluation results are shown in Table 4. We focus next on the most important model comparisons due to space constraints, e.g. we do not show results when combining two groups of features.

We started with models that included features from only one group, i.e. the individual feature group models shown in rows 1-4 in Table 4, selected the best such model and then added, sequentially, features from the other groups in batches, where each batch contained the selected features in one group. This procedure, also known as an ablation study in machine learning, allows to see what we gain if we add a group of features to a model that already contains feature from one or more groups. From Table 4, we infer that the ES and Coh-Metrix individual models are the best as they have slightly higher accuracy in prediction (85.23% for ES and 85.23% for Coh-Metrix) compared to other two individual feature groups. Also their kappas are the highest among the models with only one group of features.

In row 5, we show the results when combining all general text analysis features: ES, LIWC, and Coh-Metrix. As already mentioned before, we are directly interested in comparing the domain expert-driven model, derived from the DE features, with the model in row 5 that includes all the general text analysis features from the ES, LIWC, and Coh-Metrix groups. As we notice, these two qualitatively different models have very similar performance across all performance measures.

In addition to developing the above models from subsets of features, we used ensembles of 3 individual and combined models, respectively, in conjunction with a majority voting mechanism. For instance, if 2 or 3 out of 3 models predicted a justification as *accepted* then the final prediction for the instance was *accepted*. We experimented with voting in two different ways: (1) we used the best 3 models from the individual or combined groups of features; (2) we used the weakest 3 models obtained with any combinations of features from individual and combined groups of features; this latter case is based on results from statistics that show that combining weak classifiers should result, in general, in better performance relative to the performance of each of the weak classifiers. Both types of ensembles (weakest versus best) yielded in the best cases similar accuracies of ~86% and similar performance across all the other performance measures. The false positive rate of the weakest combined model ensemble was lowest.

5. CONCLUSIONS

In this paper, we experimented with multiple models designed to automatically assess notebook sections from engineering virtual internships. In particular, we developed models to assess notebook sections in which students justified design decisions. All models performed very well with good and very good kappa scores (kappas scores of 0.6-0.8 are considered very good)

Table 3. Descriptions of the some features used in the proposed models (not all shown due to space constraints).

Features	Description
	LIWC
Word Count	*Word Count* (WC; Total number of words in text), *Token Count* (TC; Number of unique words in text), *Words > 6 letters* (SIXLTR: total number of words greater than 6 letters) *Punctuations*
Type Token Ratio	*Ratio of TC and WC*
	Coh-Metrix
Lexical Component Counts	*DESPC* - Paragraph count, number of paragraphs; DESSC - Sentence count, number of sentences, DESWC - Word count, number of words
DESPL	DESPL - Paragraph length, number of sentences, mean; DESPLd - Paragraph length, number of sentences, standard deviation; DESSLd; Sentence length, number of words, standard deviation;
Connectives Features	PCCONNp - the degree to which the text contains connectives such as adversative, additives and comparative connectives to express relations in the text.
Temporality Features	PCTEMPz - the temporality such as tense or aspect of the text; SMTEMP - temporal cohesion, measured by repetition score of tense and aspect
LDTTRa	Type token ratio of all words.
	Domain Expert (DE)
Exoskeleton Design Inputs	Control Sensor, Range of Motion, Power Source, Material, Actuator
Dialyzer Design Inputs	Process, Surfactant, Material, Carbon Nanotube Percentage
Attributes	Referencing any design attribute or performance parameter such as cost, reliability, etc.
Justification Features	*Balancing* - Justifying input choices by stating it made up for the weakness of another choice or by saying that another choice will balance out its weaknesses; *Client* - Justifying input choices by stating it would be good for the client or end user of the product; *Consultant.Requests* - Justifying input choices because the results meet or are expected to meet internal consultants' requests; *Evaluation* - Justifying input choices by evaluating the performance of the inputs
	Essay Scoring (ES)
Token Count	Count of word occurrences in the justification.
Content Word Count	Count of all content words (noun, adjective, verb, adverb) in the justification.

Table 4. Performance evaluation results for various models.

S.N.	Features	Accuracy	FP Rate	Precision	Recall	F-Measure	Kappa
1	ES	85.2349	0.2490	0.850	0.8520	0.8510	0.6181
2	LIWC	83.2215	0.2950	0.8270	0.832	0.8290	0.5591
3	Coh-Metrix	85.2349	0.2950	0.8480	0.8520	0.8460	0.5991
4	DE	83.2215	0.3020	0.8270	0.8320	0.8280	0.5555
5	ES+LIWC+Coh-Metrix	83.8926	0.2920	0.8340	0.8390	0.8350	0.5733
6	LIWC + DE + Coh-Metrix + ES	81.8792	0.3000	0.8150	0.8190	0.8170	0.5314

indicating that they are much better than chance predictions. Our results show that, in this context, the predictive value of models using only the general text analysis features is comparable to the predictive value of a model using only the DE features (a McNemar’s test on paired nominal data revealed no significant difference between the two models’ prediction).

In particular, the ES group of features is the best predictor of students’ justifications quality. When other groups of features are added to the individual ES model, the results do not improve significantly. The fact that the ES features are so good is not surprising. Word count, or essay length, which is one of the features in the ES group, is known as being the best predictor of essay quality in automated essay grading [6,10]. Also, the Coh-Metrix group of features are a good predictor of the quality of students’ justifications.

It is important to note, however, that the predictive power of a model is only one dimension for evaluating the utility of automated assessment models in learning environments like virtual internships. We suggest that developmental cost and interpretability of the models are also valuable dimensions to consider. Of the models presented above, those using only the general text analysis features have the lowest developmental cost. Moreover, these features are generally applicable across types of tasks, specific tasks, and domains. In contrast, models containing the DE features have a relatively high developmental cost because their features required the time and expertise of humans to develop. We do note that the DE features described in this paper were automated. Thus, they can readily be applied to more justification sections from engineering virtual internships. However, these DE features are specific to this context and are likely not generalizable outside of engineering virtual internships.

The utility of these automated assessment models lies in implementing them in real-time during a virtual internship where they will be used to assess student work and either generate automatic feedback or suggest feedback for human mentors to give. For the models using only the general text analysis features, any potential feedback would be in terms of features such as word count or "narrativity" of the text that are not directly related to the domain-relevant content of the text. Those models using DE features, however, could potentially generate domain-relevant feedback in terms of what DE features were present and absent in the text. For example, if a student's justification section fails to relate their design decisions to the requests of the company's internal consultants, that is, it lacks the "Consultant Requests" DE feature, feedback could be suggested to the mentor or provided automatically to the student informing them of this missing information and suggesting ways to include it. Thus, in terms of ease of interpretation, those models using only the general text analysis features have a relatively low ease of interpretation compared to those models that include the DE features.

In this context, we then suggest the use of the best predictive model to assess the overall quality of justifications in engineering virtual engineering internships, and subsequently use the DE-based model to identify potential domain-specific missing parts in an unacceptable justification in order to provide direct feedback to the student or at least make suggestions to human mentors regarding possible weak aspects of the justification. This approach balances the tradeoffs between generality and reliability versus domain and task specific diagnostic capabilities.

We plan to further improve the predictive power, generality, and diagnostic capabilities of our models. For instance, we are considering unsupervised methods to automatically detect domain specific codes that could be used as features in our DE models. Furthermore, we are considering unsupervised topic detection in student-generated justification as a way to generalize the applicability of our models to other domains and types of tasks.

6. ACKNOWLEDGMENTS

This work was funded in part by the National Science Foundation (DRL-0918409, DRL-0946372, DRL-1247262, DRL-1418288, DUE-0919347, DUE-1225885, EEC-1232656, EEC-1340402, REC-0347000), the MacArthur Foundation, the Spencer Foundation, the Wisconsin Alumni Research Foundation, and the Office of the Vice Chancellor for Research and Graduate Education at the University of Wisconsin-Madison. The opinions, findings, and conclusions do not reflect the views of the funding agencies, cooperating institutions, or other individuals.

7. REFERENCES

[1] Chesler, N. C., Ruis, A. R., Collier, W., Swiecki, Z., Arastoopour, G., & Shaffer, D. W. (2015). A novel paradigm for engineering education: Virtual internships with individualized mentoring and assessment of engineering thinking. *Journal of biomechanical engineering*, *137*(2).

[2] Dikli, S. (2006). An Overview of Automated Scoring of Essays. *Journal of Tech., Learning, and Assessment*, 5(1).

[3] Glaser, B. G., & Strauss, A. L. (2009). The discovery of grounded theory: Strategies for qualitative research. Transaction Publishers.

[4] Graesser, A. C., McNamara, D. S., Louwerse, M. M., and Cai, Z. (2004). Coh-Metrix: Analysis of text on cohesion and language. *Behavior research methods, instruments, & computers*, 36, 2(2004), 193-202.

[5] Leacock, C., and Chodorow, M. 2003. C-rater: Automated scoring of short-answer questions. *Computers and the Humanities*, 37, 4(2003), 389-405.

[6] Mohler, M., and Mihalcea, R. 2009. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the EACL* (Athens, Greece, March, 2009).

[7] Pennebaker, J. W., Francis, M. E., and Booth, R. J. 2001. Linguistic inquiry and word count: LIWC 2001. *Mahway: Lawrence Erlbaum Associates*, 71, (2001), 2001.

[8] Rus, V., Feng, S., Brandon, R., Crossley, S., and McNamara, D.S. (2011). A Linguistic Analysis Of Student Generated Paraphrases. In R. C. Murray and P.M. McCarthy (Eds.), Proceedings Of The 24th International Florida Artificial Intelligence Research Society Conference. Menlo Park, CA: AAAI Press.

[9] Rus, V., Lintean, M., Azevedo, R. (2009). Automatic Detection of Student Mental Models During Prior Knowledge Activation in MetaTutor. Proceedings of the 2nd International Conference on Educational Data Mining. Cordoba, Spain.

[10] Rus, V., Niraula, N. (2012). Automated Detection of Local Coherence in Short Essays Based on Centering Theory", CICling 2012, March 11-17, IIT Delhi, India.

[11] Shaffer, D. W. (2006). *How computer games help children learn*. Macmillan.

[12] Shaffer, D.W., Borden, F., Srinivasan, A., Saucerman, J., Arastoopour, G., Collier, W., Ruis, A.R., & Frank, K.A. (2015). The *nCoder*: A technique for improving the utility of inter-rater reliability statistics. Epistemic Games Group Working Paper 2015-01. University of Wisconsin–Madison.

[13] Shermis, M.D. & Burstein, J. (2003). Automated Essay Scoring: A Cross Disciplinary Perspective. Lawrence Erlbaum Associates, Mahwah (2003).

[14] Tisi, J., Whitehouse, G., Maughan, S., & Burdett, N. (2013). *A review of literature on marking reliability research* (Report for Ofqual). Slough: NFER.

[15] Williams, C., &D'Mello, S. (2010). Predicting student knowledge level from domain-independent function and content words. In Intelligent Tutoring Systems (pp. 62-71). Springer Berlin Heidelberg.