The Thirty-Second International Florida
Artificial Intelligence Research Society Conference (FLAIRS-32)

Effect of Domain Corpus Size and LSA Vector Dimension: A Study in Assessing Student Generated Short Texts in Virtual Internships without Participant Data

Dipesh Gautam, Zhiqiang Cai, Vasile Rus
Department of Computer Science and Institute for Intelligent Systems,
The University of Memphis, Memphis, TN, 38152
dgautam@memphis.edu

Abstract

Semantic similarity is a major automated approach to address many tasks such as essay grading, answer assessment, text summarization and information retrieval. Many semantic similarity methods rely on semantic representation such as Latent Semantic Analysis (LSA), an unsupervised method to infer a vectorial semantic representation of words or larger texts such as documents. Two ingredients in obtaining LSA vectorial representations are the corpus of texts from which the vectors are derived and the dimensionality of the resulting space. In this work, we investigate the effect of corpus size and vector dimensionality on assessing student generated content in advanced learning systems, namely, virtual internships. Automating the assessment of student generated content would greatly increase the scalability of virtual internships to millions of learners at reasonable costs. Prior work on automated assessment of notebook entries relied on classifiers trained on participant data. However, when new virtual internships are created for a new domain, for instance, no participant data is available a priori. Here, we report on our effort to develop an LSA-based assessment method without student data. Furthermore, we investigate the optimum corpus size and vector dimensionality for these LSA-based methods.

Introduction

Semantic similarity is about determining whether two texts (documents, paragraphs, or words) are similar in their meaning. Often the semantic similarity methods represent the documents or terms using a vectorial representation and then apply a similarity function such as computing the cosine of the angle between the corresponding vectors of the documents. The cosine is equivalent to the normalized dot product of the two vectors thus quantifying to what degree the two vectors are close to each other. The similarity score obtained with such methods depends upon the vectors used in the calculation. The vectors are derived based on a statistical analysis of a large corpus of documents. The analysis produces a term-by-document matrix in which terms represent the rows and documents the columns. Each cell in the matrix indicates, for instance, how frequent the corresponding term in the row is in the corresponding document in the column. Latent Semantic Analysis (LSA) uses Singular Value Decomposition (SVD) to map such high-dimensionality term-by-document matrices onto reduced-rank matrices with the added benefit of being able to capture second order semantic relationships among words (Landauer, Foltz, and Laham 1998; Bradford 2008).

Studies have shown that the rank (number of dimensions) of the LSA semantic space and therefore of the LSA vectors (Landauer, Foltz, and Laham 1998; Bradford 2008) as well as the corpus size and its nature (Kontostathis 2007; Crossley, Dascalu, and McNamara 2017) influence the quality of the resulting vector representations. Landauer et al. (Landauer, Foltz, and Laham 1998) noted that a vector dimension of 300 obtained from moderate corpus sizes performs best in general. However, obtaining the optimum corpus size and vector dimensionality for a particular domain of interest and task is yet to be determined. In this study, we investigate the optimum domain corpus size and vector dimensionality to develop LSA based assessment methods for virtual internships.

Background

Virtual Internships and Automated Assessment

Virtual internships are online simulations where students experience professional practices similar to actual internships (Shaffer 2006). During virtual internships, students work on various tasks, e.g., engineering design tasks. While working on these tasks, they need to provide justifications for their work, e.g., justifications for their designs, in digital notebooks. The notebook entries are then assessed as acceptable or unacceptable by human raters. Automating this assessment task is an important step towards reducing the time and cost associated with developing and running such virtual internships. Previous attempts to automatically assess notebook entries in virtual internships focused on automated classifiers trained on participant data collected after the initial development and deployment of the corresponding virtual internship. This means that one needs to run the virtual internship at the beginning using human graders, which is tedious, time-consuming, not-scalable, and expensive. The major challenge to developing automated assessment methods from the very start is the fact that participant data is not available at the time when a new internship is being developed, e.g., for a completely new set of tasks or a completely new domain(Swiecki and Shaffer

2017)). For instance, when instructors create new activities or customize existing activities for an existing internship, previously trained assessment classifiers become invalid. Indeed, customization makes trained clasifiers invalid whereas new internships for new domains leaves the virtual internship system without classifiers until learner data is collected. Our work is motivated by this need to develop automated assessment method early on, at design time, for a new virtual internship when participant data is not yet available. To this end, we explore methods to generate classifiers without student data. Such approaches would enable the development of virtual internships that incorporate automated assessment methods from the very beginning, avoiding the need to have human raters assess learner responses during initial deployment of such new virtual internships. We further extend prior work (Gautam et al. 2017) by exploring optimum corpus size and the LSA vector dimensionality for student notebook assessment.

LSA and its Uses in Text Analysis

LSA is a vector space model for deriving semantic representations of words and largers texts such as documents. It relies on a reduced-rank approximation of a term-document matrix which captures word co-occurrence information from natural texts such as textbooks. The rank reduction is used to remove the noise introduced by sparsity of the term document matrix. Jessup and Martin (Jessup and Martin 2001) showed that the optimal rank choice delivers improvement in performance in an information retrieval task. Using LSA vectors instead of using standard term frequency - inverted document frequency (tf-idf) vectors in a retrieval system, the system will be able to retrieve documents based on concepts as opposed to keywords, which improves the recall of the system(Bellegarda 2005). It should be noted that, usually, when recall increases precision decreases.

In addition to the rank (vector dimension), the size of the input corpus also has an impact on the performance of LSA vector spaces. A study conducted by Crossley et al. (Crossley, Dascalu, and McNamara 2017) showed that LSA space developed using larger corpora performed better than when smaller corpora was used in a word association task and a vocabulary level test. It should be noted that they developed the LSA space using a multiple domain corpus (TASA[1]) and a single domain corpus from the Corpus of Contemporary American English (COCA) (Davies 2010). Furthermore, Landauer et al.(Landauer, Foltz, and Laham 1998) generated an LSA space from encyclopedia articles and used it for a Test of English as a Foreigh Language (TOEFL[2])word comparison task. They found that the vectors with dimensionaliy of 300 performed best, which was considered a standard number of dimensions for many applications.

LSA has been used successfully in various tasks such as answer grading, text summarization, e-mail categorization, and information retrieval. For instance, LSA based essay grading yields comparable performance to human graders. Landauer and colleagues compared the human ratings of passages written by students with LSA-obtained rating and found that the meaning of passages could be carried by the words independent of their order, which is what LSA is about (Landauer et al. 1997). In another study, (Mohler and Mihalcea 2009) showed that an LSA based short answer grading method performed as well as a knowledge based method, e.g., methods that rely on knowledge-based resources such as WordNet. The LSA based approach has a major advantage over the knowledge based approach - the LSA model could be constructed automatically, in an unsupervised manner, whereas knowledge based approaches, e.g., such as those relying on WordNet, require extensive manual efforts to build the knowledge based resources.

Pérez et al (Pérez et al. 2005a; 2005b) obtained LSA space from a large collection of pre-categorized domain corpus and combined with the BLEU algorithm, a method used to evaluate machine translation (Papineni et al. 2002), to assess student's freely generated textual answers. They claimed that their method achieved state-of-the-art correlations to teachers' scores. In their method, they averaged word vectors to represent student and reference answers. It should be noted that, though our vector representation approach is similar, we have collected domain corpus from Wikipedia by automatically filtering the Wikipedia articles.

Other uses of LSA relevant to our work are from text summarization. LSA based methods have been shown to extract better summaries (Gong and Liu 2001; Steinberger and Ježek 2004; Yeh et al. 2005), particularly, when choosing appropriate vector dimensions, LSA helps to extract better semantically similar summaries from original documents when compared to keyword based summaries (Yeh et al. 2005).

Another related method was proposed by Dredze and colleagues (Dredze et al. 2008) to generate keywords for e-mail messages without annotated training data. E-mail summary keywords are generated to represent e-mails in e-mail filtering systems. The keywords act as features for e-mail classification. In the Dredze and colleagues' method, they generated LSA vectors to represent each word in e-mails and identified each word as a keyword if the word was present in an e-mail. They claimed that the LSA based method provided a good representation of e-mails compared to tf-idf based approaches.

Like these approaches above, our method uses LSA for assessing learners' free responses in learning environments such as virtual internships. We explore the impact of the corpus size and space dimensionality on the performance of the resulting assessment method. This work leverages the previous approach (Cai et al. 2018) of extracting domain corpus. However, it should be noted that the the previous work analyzed the learners' responses that were few words to a sentence length. Whereas in this work we further extend previous study by analyzing the learners' response consisting of a paragraph of two to four sentences.

Our Method

Our method involves a two step process. First, we develop core concept identifiers based on a small set of seed data provided by teachers when developing a new internship or

[1] http://lsa.colorado.edu/spaces.html

[2] www.ets.org/toefl

new tasks for an exiting internship. Second, we evaluated the performance of these classifiers as a function of the corpus size and dimensionality of the semantic space. We describe next the method and the corpus we used in our experiments.

Classifier Generation

In the first step of our method for generating classifiers without student data, teachers define a small set of seed concepts for a given task, e.g., an engineering design task. The seed concepts represent the key semantic content student notebook entries should contain. In addition, teachers provide a small set of exemplar or benchmark notebook entries in which they tag the seed concepts.

Table 1: Algorithm to obtain average similarity score between the chunks of a concept in exemplars.

***Input**: Set C of annotated chunks for a concept in exemplars*
***Output**: Average A and standard deviation SD of similarities between chunks of a concepts in exemplars*
Initialize: S = empty list of similarities
Initialize: P = empty list of set of chunk pairs
do for each c_i in C:
  do for each c_j in C:
    p = set(c_i,c_j)
    if $c_i \neq c_j$ and p not in P:
      S = S + similarity(c_i,c_j)
      P = P + p
A = Average(S)
SD = Standard deviation(S)

Using this seed information we develop a semantic similarity based identifier for each seed or core concept. That is, we try to identify whether a target core concept is present in the student answer or not. There is one identifier for each of the core concepts. The identifiers are modeled as classifiers. Each of the classifiers uses a sliding window to search for the chunk of text in a participant notebook entry that is most similar to the target core concept. For this purpose, we use a sliding window of size equal to the length of the core concept. The window slides over the participant response and for each instance, we calculate the similarity between the chunk of text in the window and a target core concept. We select the chunk of text in the student notebook entry that corresponds to the highest similarity score.

If the highest score is higher than a threshold, we decide that the target core concept is present in the student response. In order to obtain the threshold for a target core concept, we calculated similarity scores between all possible tagged chunk pairs for that concept in the teacher generated ideal responses, ie., exemplars. Then, we computed the threshold as one standard deviation below the average similarity score. The detailed steps for obtaining the average and standard deviation are provided in Table 1.

To calculate the similarity score between core concepts and the sliding window, we use an LSA-based semantic similarity implementation available in SEMILAR(Rus et al. 2013). Figure 1 shows the overview of obtaining cosine similarity

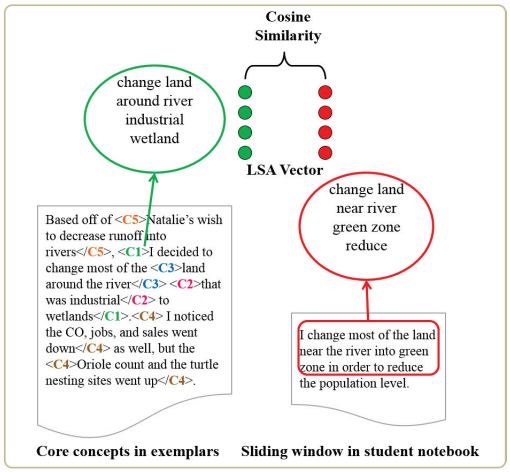

Figure 1: Concepts in examplers and sliding windows.

scores between a core concept and the chunk of texts corresponding to one instance of the sliding window.

In our previous work (Rus et al. 2016), we obtained word vectors from the TASA corpus. The corpus consists of contents from textbooks, literature works. This domain general LSA model was developed from about 38,000 documents and 92,000 terms (Stefănescu, Banjade, and Rus 2014). We compared the performance of our classifiers using the TASA corpus to the performance obtained with a domain specific corpus, which is described next.

Domain Corpus Collection

To collect the domain specific corpus, we started with a seed corpus of a small number of documents from our target internship, *Land Science*. These documents include text resources about urban planning and instructions sent to participants during the virtual internship. Next, we extracted keywords from this seed corpus and assigned a "keyness" value to each keyword. The keyness depends upon two factors: first, if a word occurs frequently in domain general corpus (such as TASA), then the word is less important for a particular domain. Second, if a word occurs frequently in a domain specific corpus, the word is important for that domain. Hence the keyness value is obtained by taking into account both factors as described in (Cai et al. 2018). An average "keyness" value is obtained based on the keyness values for each word that appears in a Wikipedia document. This average keyness is viewed as the document keyness by which the documents are ranked in order to select the domain specific documents.

While LSA vectors are essential components in our method, the dimensionality of the LSA space and therefore of the LSA vectors as well as the size of the domain specific corpus are important parameters that affect the predictive power as well as scalability of the classifiers we develop. Therefore, in this work we analyze the impact of these parameters on the performance of the assessment classifiers by exploring different corpus sizes and vector dimensionalities and observing their impact on the performance of our classifier in terms of F-1 scores, as explained next.

Experiments and Results

Dataset

As mentioned, our goal is to study how corpus size, corpus domain specificity, and dimensionality of LSA vectors affect the performance of student answer assessment methods. Our experimental data consists of student responses in virtual internships which were used to evaluate our classifiers. The classifiers classify each sentence from student responses as the presence or absence of a core concept in it.

Table 2: Annotation of exemplar for core concepts.

Based off of <C5>Natalie's wish to decrease runoff into rivers</C5>, <C1>I decided to change most of the <C3>land around the river</C3><C2>that was industrial</C2>to wetlands</C1>. <C4>I noticed the CO, jobs, and sales went down</C4>as well, but the <C4>Oriole count and the turtle nesting sites went up</C4>.

Table 3: Number of core concepts in exemplars and student notebooks.

Concepts	Notation	#_E	#_S
Runoff	C1	2	26
Phosphorous	C2	2	15
Orioles	C3	3	25
Housing	C4	4	33
Turtles Nesting Sites	C5	3	17
Jobs	C6	5	45
Sales	C7	2	27
CO	C8	4	17
Justification	C9	11	101
Land use change	C10	13	120
Indicator change	C11	14	92
Indicator value	C12	9	5
Directly quotes information from resource readings or teammate	C13	5	0

Note: #_E for exemplars and #_S for student notebooks.

i Domain corpus: It consists of documents selectively chosen from Wikipedia articles. The corpus consists of 32,000 documents that are relevant to the Land Science virtual internship. From the collection, we randomly selected 2,000, 4,000, 8,000, 16,000 and 32,000 documents to generate 1,000 dimension LSA word vectors. Since the vector dimensions are ranked, we generated 1,000 dimension vectors, instead of generating separate LSA vectors of different dimensions with varying number of documents. Selecting fewer dimensions from 1,000 dimension vector simplifies the LSA vector generation process with negligible information loss.

ii Notebooks: Two sets of of annotated notebooks from Land Science, one with 14 exemplars created by teachers and 100 randomly selected participant notebooks entries.

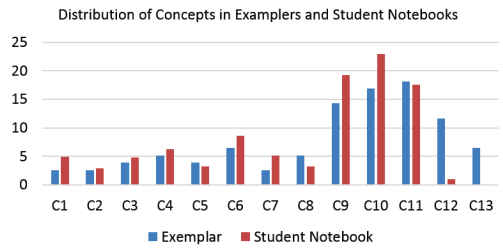

Figure 2: Distribution of core concepts.

The exemplar notebooks were annotated for the core concepts (see Table 2). The 100 notebook entries were split into 550 sentences. These sentences are manually filtered to remove noisy sentences that only consist of non-english words or out of domain words. After filtering, we were left with 278 sentences, each of was then annotated with the presence or absence of the 13 core concepts. It should be noted that a notebook may consist of a small subset of core concepts, which means some concepts are more likely to appear than the others as seen in Table 3. The distribution of concepts in exemplars and student notebooks is seen in Figure 2. From the figure, it can be noted that some of the concepts (e.g., C12 and C13) are much more rare in student answers compared to the exemplars. For performance calculation, we do not include concepts (such as C13) which are absent in student notebook.

Besides the domain specific corpus, we also used the TASA corpus in order to compare the performance with the domain corpus.

Results

Figure 3 shows the surface plot of average F-1 scores for all core concepts for different combinations of domain corpus size (Spaces) and number of space dimensions (Dimensions). The plot suggests that the performance initially improves as the corpus size increases, up to a certain point, after which the performance starts decreasing. Furthermore, the figure indicates that the performance initially improves with an increasing number of dimensions, stays constant for a little while, then starts decreasing and finally increases again. It could be seen that the overall performance is determined by a combination of corpus size and vector dimensionality.

Table 4: Minimum and maximum average performance among core concepts for the combination of corpus size and vector dimension.

	Metrics	Values	(Space, Dimension)
Minimum	P	0.798	(4000, 1000)
	R	0.600	(32000, 2)
	F	0.620	(32000, 2)
Maximum	P	0.925	(32000, 1000)
	R	0.701	(32000, 1000)
	F	0.716	(32000, 1000)

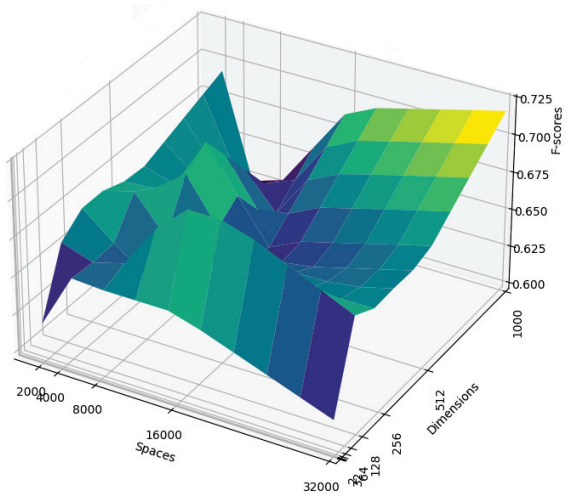

Figure 3: Surface plot of average F-1 scores for domain corpus of different sizes (spaces) and number of dimensions.

In the Table 4, the minimum and maximum of Precision, Recall, and F-1 scores along with the corresponding combination of corpus size and vector dimensionality for the domain corpus are shown. These scores are obtained by averaging the corresponding metrics for all 13 concepts. From the table, we see that 1,000 dimensions with 32,000 documents performed best(F-1=0.71). It should be noted, however, that even though the performance is better compared to other combinations, other combinations with smaller corpus sizes and fewer dimensions perform comparably well (see Figure 5). Figure 5 further suggests that the F-1 score initially improves (except for a corpus size of 16,000) when vector dimensionality increases, followed by a drop, and then improves again at a quick rate followed by a plateau in which the performance remains constant. Moreover, it is seen from the heatmap (Figure 4) that the performance remains comparatively same with a corpus size of 4,000 through 32,000 documents for dimensionality of 32. The overall performance is almost comparable to the maximum performance (F-1=0.716), suggesting that a small corpus (size=400) and small dimensionality (=32) could be good enough to develop an assessment component for virtual internships.

Figure 6 shows the trend of F-1 scores for varying vector dimensions of the TASA corpus. The graph resembles the similar pattern observed for the domain specific corpus, where the performance improves initially, then remains comparatively the same and then starts dropping again. It can be concluded that a small domain corpus can give better performance than the much larger TASA corpus when a proper combination of corpus size and vector dimensionality is chosen.

Conclusion and Future Works

We presented in this paper an approach to develop student answer assessment methods without student data. We also analyzed the impact of corpus size, corpus specificity, and semantic space dimensionality on the performance of the assessment methods. Our analysis showed that the LSA spaces generated from a domain specific corpus can perform better when compared to a space generated from the much larger TASA corpus for the notebook assessment task. For the domain specific corpus, the best average performance over all the target concepts was obtained for the maximum available corpus size and the maximum number of dimensions. However, this performance is comparable to results obtained with a smaller corpus size and smaller vector dimensionality indicating that smaller corpora and spaces can be good enough to boost assessment components for virtual internships. We plan to further experiment with our method on other virtual internships and data from other adaptive learning technologies to see if our current conclusions hold on such new data.

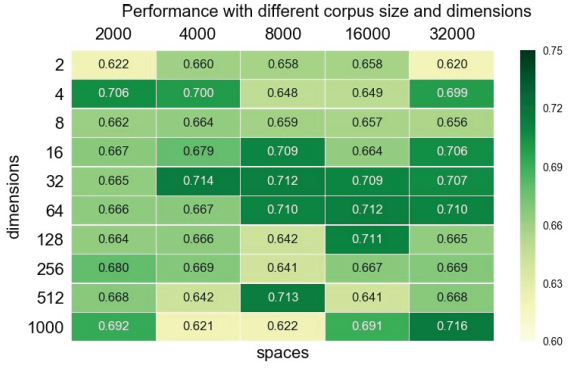

Figure 4: Heatmap for F1 scores with different combinations of corpus size and vector dimensions.

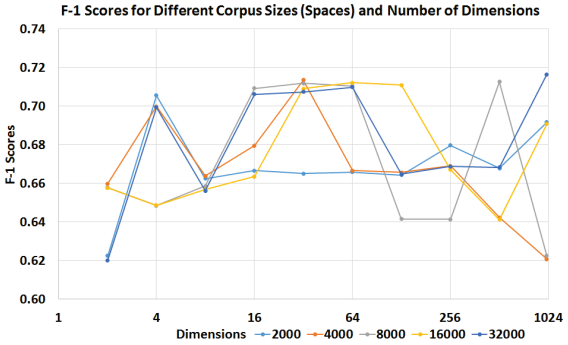

Figure 5: Average F-1 scores for domain corpus of different sizes (see legends in the figure) and number of dimensions.

Acknowledgments

This work was partially supported by The University of Memphis, the National Science Foundation (awards CISE-IIS-1822816 and CISE-ACI-1443068), and a contract from the Advanced Distributed Learning Initiative of the United States Department of Defense.

References

Bellegarda, J. R. 2005. Latent semantic mapping [information retrieval]. *IEEE signal processing magazine* 22(5):70–

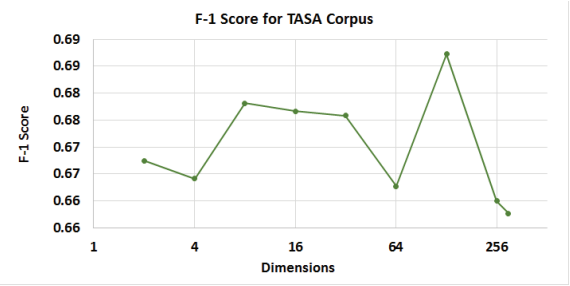

Figure 6: F-1 scores for TASA corpus with varying dimensions.

80.

Bradford, R. B. 2008. An empirical study of required dimensionality for large-scale latent semantic indexing applications. In *Proceedings of the 17th ACM conference on Information and knowledge management*, 153–162. ACM.

Cai, Z.; Graesser, A.; Windsor, L.; Cheng, Q.; Shaffer, D. W.; and Hu, X. 2018. Impact of corpus size and dimensionality of LSA spaces from wikipedia articles on autotutor answer evaluation. In *Proceedings of the 11th International Conference on Educational Data Mining, EDM 2018, Buffalo, NY, USA, July 15-18, 2018*.

Crossley, S. A.; Dascalu, M.; and McNamara, D. S. 2017. How important is size? an investigation of corpus size and meaning in both latent semantic analysis and latent dirichlet allocation. In *30th International Florida Artificial Intelligence Research Society Conference, FLAIRS 2017*. AAAI Press.

Davies, M. 2010. The corpus of contemporary american english as the first reliable monitor corpus of english. *Literary and linguistic computing* 25(4):447–464.

Dredze, M.; Wallach, H. M.; Puller, D.; and Pereira, F. 2008. Generating summary keywords for emails using topics. In *Proceedings of the 13th International Conference on Intelligent User Interfaces*, IUI '08, 199–206. New York, NY, USA: ACM.

Gautam, D.; Zachari Swiecki, D. W.; Graesser, A. C.; and Rus, V. 2017. Modeling classifiers for virtual internships without participant data. 278–283.

Gong, Y., and Liu, X. 2001. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 19–25. ACM.

Jessup, E., and Martin, J. 2001. Taking a new look at the latent semantic analysis approach to information retrieval. *Computational information retrieval* 2001:121–144.

Kontostathis, A. 2007. Essential dimensions of latent semantic indexing (lsi). In *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on*, 73–73. IEEE.

Landauer, T. K.; Laham, D.; Rehder, B.; and Schreiner, M. E. 1997. How well can passage meaning be derived without using word order? a comparison of latent semantic analysis and humans. In *Proceedings of the 19th annual meeting of the Cognitive Science Society*, 412–417.

Landauer, T. K.; Foltz, P. W.; and Laham, D. 1998. An introduction to latent semantic analysis. *Discourse processes* 25(2-3):259–284.

Mohler, M., and Mihalcea, R. 2009. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, 567–575. Association for Computational Linguistics.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, 311–318. Association for Computational Linguistics.

Pérez, D.; Alfonseca, E.; Rodríguez, P.; Gliozzo, A.; Strapparava, C.; and Magnini, B. 2005a. About the effects of combining latent semantic analysis with natural language processing techniques for free-text assessment. *Revista signos* 38(59):325–343.

Pérez, D.; Gliozzo, A. M.; Strapparava, C.; Alfonseca, E.; Rodriguez, P.; and Magnini, B. 2005b. Automatic assessment of students' free-text answers underpinned by the combination of a bleu-inspired algorithm and latent semantic analysis. In *FLAIRS conference*, 358–363.

Rus, V.; Lintean, M.; Banjade, R.; Niraula, N.; and Stefanescu, D. 2013. Semilar: The semantic similarity toolkit. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 163–168.

Rus, V.; Gautam, D.; Swiecki, Z.; Shaffer, D. W.; and Graesser, A. 2016. Assessing student-generated design justifications in virtual engineering internships. In *EDM*, 496–501.

Shaffer, D. W. 2006. *How computer games help children learn*. Macmillan.

Stefănescu, D.; Banjade, R.; and Rus, V. 2014. Latent semantic analysis models on wikipedia and tasa. In *Language Resources Evaluation Conference (LREC)*.

Steinberger, J., and Ježek, K. 2004. Using latent semantic analysis in text summarization and summary evaluation. In *In Proc. ISIM '04*, 93–100.

Swiecki, Z. M., and Shaffer, D. 2017. Dependency-centered design as an approach to pedagogical authoring. *Game-based learning: Theory, strategies and performance outcomes*.

Yeh, J.-Y.; Ke, H.-R.; Yang, W.-P.; and Meng, I.-H. 2005. Text summarization using a trainable summarizer and latent semantic analysis. *Information processing & management* 41(1):75–95.