Using vector space models based on LDA and LSA for automatically grading exam questions

Johanna de Vos, U908153, TxMM, 19/01/2018

ABSTRACT

Students' answers to open-ended exam questions can be automatically graded by comparing them to a 'perfect' reference answer in terms of semantic similarity. In this project, I compared three methods for constructing a vector space to measure semantic similarity in. The dimensions of this vector space were made up of: 1) topics, generated by LDA models, 2) singular values, computed by LSA models, and 3) vocabulary counts, serving as baseline models. Better results were expected for the LDA and LSA models, as they capture students' answers at the semantic level rather than depending on exact vocabulary matches. Indeed, the best result was found for one of the LDA models, which achieved a Spearman's correlation coefficient of .52 between the grades predicted by the model and the true grades assigned by the lecturer. The LSA models on average outperformed the baseline models as well. In addition, it was found that the type of data used for training the LDA and LSA models impacted the accuracy of the predictions.

KEYWORDS

Automatic grading, LDA, LSA, Semantic similarity, Text mining, Topic models, Vector space models

1 INTRODUCTION

Methods for automatically grading answers to exam questions have been developed since the 1960s. Initially, these approaches were based on structural features of student answers, such as the total number of words and the average sentence length (e.g., [1]). In recent years, researchers' focus has shifted to content-based approaches. One such research avenue is based on the concept of semantic similarity: quantifying to what extent two documents are alike in terms of the meaning that they express.

For example, one can compare how similar a student's answer to an exam question is to the 'perfect' answer to this question [2], which I will call the reference answer. This is the approach that I will follow in the current study. The basic idea is that that better student answers will resemble the reference answer more closely than not-as-good student answers. Measures of semantic similarity can be used to quantify how closely a student answer resembles the reference answer: high semantic similarity indicates that the student answer deserves a high grade.

The semantic similarity between two documents (here: a student answer and the reference answer) can be quantified by calculating the cosine of the angle between the two vectors that represent these documents in a semantic space. How the dimensions in this semantic space come about depends on the underlying model that is used to capture the meaning of a document. In this project, I will focus on two such models: Latent Dirichlet Allocation (LDA) and Latent Semantic Analysis (LSA).

LDA models [3] are generative topic models that take the form of a probability distribution over topics in a document collection. In turn, each topic takes the form of a probability distribution over words. LDA models are calculated from a document-term matrix. When using such a model to calculate semantic similarity between two documents, each dimension in the semantic space represents one topic. More specifically, for each document its vector representation consists of percentages that reflect the proportion of the document concerned with each topic. For example, if we have an LDA topic model with three topics ['Animals', 'Science', 'Sports'], the vector for a particular document could be [70, 23, 7].

LSA models [4] are also topic models that are calculated from a document-term matrix, but they are conceptually and mathematically very different from LDA models. LSA reduces the dimensionality of the document-term matrix. Thus, in the semantic space, the dimensions will be those orthogonal components that are the result of the dimensionality reduction. They cannot be named or interpreted intuitively, like the topics in an LDA model. Nevertheless, we can calculate the distance between two document vectors in this new vector space to obtain the semantic similarity between two documents.

LDA and LSA models have previously been used for grading exam questions. Almost all of these studies focus on LSA (e.g., [5]). To my knowledge, only one study [6] has employed LDA for grading exam questions, comparing its effectiveness to that of LSA. In [6], Kakkonen, Myller, Sutinen and Timonen (2008) expected LDA to outperform LSA, citing earlier information retrieval studies in which this was the case, but found the opposite. They explain this may be due to the small size of their training data (26, 42 and 70 student essays in three experiments).

The current study is motivated by the scarcity of studies on the use of vector space models in automatic grading in general, as well as by the question of whether Kakkonen et al.'s (2008) training set impacted their results. To investigate the latter issue, I will make use of two different datasets for training, the first one being student answers (like in [6]), and the second one being a chapter of a psychology textbook, which is more diverse in its contents. Thus, the research questions will be the following:

1. How effective are LDA-based and LSA-based vector space models for automatically grading students' answers to open exam questions?
2. Does the effectiveness of the above models depend on the training data that is used?
3. Are these topic models more effective than a simple vector space model based on vocabulary counts?

2 APPROACH

2.1 Data

The core dataset consisted of 402 students’ written answers to a first year Psychology exam question from Radboud University. The question was: “Discuss Whorf's language theory. Include the following terms in your answer: Strong and weak variations on the theory.” All answers were accompanied by the grade that the lecturer had assigned to that answer. Permission to work with these data was obtained through the ethical commission of the Social Science faculty. The student answers were used for two purposes: to train topic models (on the training set), and to predict grades from (on the test set). The total number of words was 23467.

As described in the introduction, I also used another text source to train the topic models. This was chapter 10 from [7], the introductory psychology textbook that the students used. Chapter 10 is the chapter in which Whorf’s language theory is explained. The chapter consisted of 1217 sentences (acting as documents in the document-term matrix) and a total of 15096 words.

The reference answer was based on a rubric that was provided by the lecturer, in combination with the definition of Whorf’s language theory as given in [7]. It consisted of 59 words in 7 sentences.

2.2 Preprocessing

Pre-processing of the data (student answers and textbook) was done in Python (version 3.6) with the Natural Language Toolkit (NLTK) (version 3.2.4) [8]. This included tokenization, lemmatization and stop word removal. Manual spelling correction was applied to the student answers (see 2.5.2).

2.3 Implementation

2.3.1 Baseline. The baseline models were vector space models whose dimensions corresponded to the words in the vocabulary of the test data. The way in which this vocabulary was counted is explained in 2.5.1. Thus, the baseline models were no topic models and were not trained on any data.

2.3.2 Topic models. The LDA models were instances of the *LdaModel* class in Python’s gensim library, and the LSA models were instances of the *LsiModel* class. The number of topics (in LDA) or singular values (SVs) (in LSA) needs to be specified beforehand. To find the optimal value for these hyperparameters, I performed a grid search on the training data (see 2.3.3).

2.3.3 Training the models. The student answers were split into a 80/20 train/test set. 10-fold cross-validation was applied to the training set in order to find the optimal number of topics and SVs. The models were evaluated by predicting grades for the answers in the validation subset of the training set (see 2.4). I used this optimal number of topics or SVs to train the final topic models on all of the training data.

The textbook chapter was not split into a training and test set because it was only used for training topic models, and did not contain any grades that could be predicted. Again, various topic models were trained, differing in their number of topics or SVs. These models were evaluated by predicting grades for all student answers in the training set. Later, I used the best-performing models to predict grades on the test set.

2.4 Evaluation

The performance of all models was evaluated by correlating the predicted grades with the grades that the lecturer had assigned. Spearman’s correlation coefficient (r_s) was used because the assigned and predicted grades often were not normally distributed.

2.5 Exploring Some Parameters

In addition to comparing model types (baseline vs. LDA vs. LSA) and the influence of the training data (student answers vs. textbook chapter), I also explored some other variables that can be expected to influence the accuracy of the model predictions.

2.5.1 Counting method. I compared TF-IDF with RAW counts, expecting better results for TF-IDF because it penalizes common words that do not distinguish between topics very well. TF-IDF was not implemented in the LDA models, which require integer counts due to being probabilistic models. I also looked at BINARY counting (0/1), as the student answers were relatively short and it is conceivable that the presence or absence of certain terms is already informative enough.

2.5.2 Spelling correction. If no spelling correction is applied, misspelled words will get their own vocabulary ID (e.g., ‘critizing’ will be different from ‘criticizing’), leading to an unnecessary increase in the dimensionality of the dataset. The correct entry ‘criticizing’ will have lower counts, and therefore may be assigned a less important role in the topic models than it should have in reality. Therefore, I expected better outcomes when the data were spelling-corrected.

2.5.3 Mapping algorithm. Semantic similarity between the student answer and the reference answer was measured in terms of cosine similarity, which lies between 0 and 1. The most straightforward way to transform this value into a grade is to multiply it by 10. I will call this mapping algorithm ‘X10’. However, Figure 1 below shows that the grades of 1 and 9 were almost never assigned by the lecturer, but 0 and 10 quite often. Therefore, to try to improve prediction accuracy, the second mapping algorithm will first multiply the cosine similarity by 10, but any 1s will subsequently be mapped to 0, and any 9s to 10. I will call this mapping algorithm ‘NO 1 OR 9’.

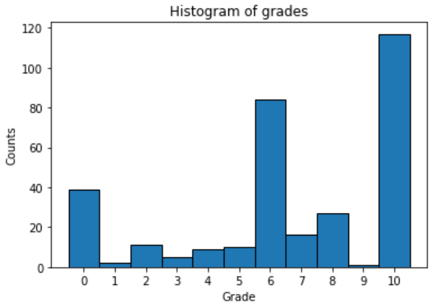

Figure 1: Frequency distribution of the grades that were assigned by the lecturer (training set only).

3 RESULTS

3.1 Number of Topics / SVs

When LDA models were trained on the student answers, those models that used only two topics on average performed best (r_s = .39), as compared to models with 4 or 7 topics. I say ‘on average’ because I averaged over the variables described in section 2.5. When training LDA models on the textbook, the average outcomes were quite different: the best-performing LDA models contained 20 topics (r_s = .49), as compared to 2, 4, 7, 12 or 40 topics.

The LSA models on average did best with 100 SVs (r_s = .38) when being trained on the student answers, as compared to 20, 50 or 200 SVs. When training them on the textbook, 10 SVs were used in the best average performance (r_s = .35), as compared to 5, 20, 50, 100, 200 or 400 SVs.

Thus, in training the final models, I set the number of topics or SVs to the above-reported numbers that had yielded the best outcomes on the two types of training data.

3.2 Outcomes per Model Type and Variable

Table 1 shows the average correlation that was obtained on the test data between predicted and lecturer-assigned grades, for each model type. The first column contains the four variables under investigation. For each variable, the results in columns 3-5 are averaged over the levels (column 2) of the other three variables. For example, the results for the two types of training data (student answers vs. textbook chapter) are averaged over the different levels of counting method, spelling correction and mapping algorithm. It is possible that there could be (an) interaction(s) between the variables, but that is outside the scope of this study.

Table 1: Correlations for all model types and variables.

		Baseline	LDA	LSA
Training data	Student answers	N/A	.36	.39
	Textbook chapter	N/A	.42	.32
Counting method	Raw	.38	.38	.34
	Binary	.34	.40	.35
	TF-IDF	.30	N/A	.41[1]
Spelling correction	Yes	.34	.37	.36
	No	.34	.41	.36
Mapping algorithm	x10	.35	.36	.36
	No 1 or 9	.33	.42	.36
Average		**.34**	**.39**	**.36**

3.3 Best-Performing Individual Models

The best-performing LDA model achieved a correlation of r_s = .52. It was trained on the textbook chapter, used binary counting, was spelling-corrected and used the x10 mapping algorithm. The scatterplot of predicted and assigned grades is shown in Figure 2.

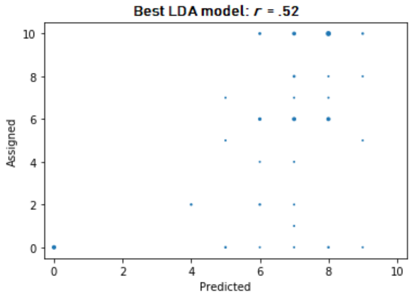

Figure 2: Scatterplot of assigned and predicted grades for the best LDA model.

The best-performing LSA model achieved a correlation of r_s = .42. It was trained on non-spelling-corrected student answers and used binary counting. The mapping algorithm was not relevant, as no 1s and 9s were predicted to begin with. Figure 3 shows the scatterplot for this model.

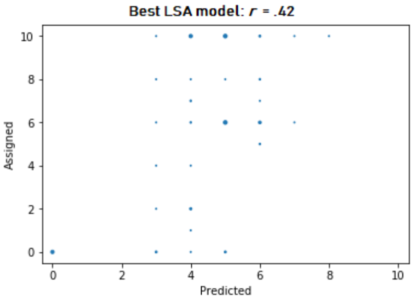

Figure 3: Scatterplot of assigned and predicted grades for the best LSA model.

Both of these models outperformed the best baseline model, which achieved a correlation of r_s = .38, using raw counts and spelling-corrected data. As in the LSA model, the mapping algorithm was irrelevant. The scatterplot is shown in Figure 4.

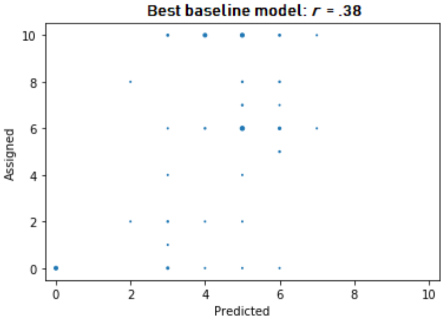

Figure 4: Scatterplot of assigned and predicted grades for the best baseline model.

These three highest correlations all were significant (all $p < .001$).

[1] Due to technical difficulties, it was not possible to implement the LSA model on the textbook chapter using TF-IDF as the method of counting. Thus, this cell specifically contains the correlation when using student answers as training data. In all other cells in the LSA column as well, the here-mentioned combination of variable levels is missing from the average.

4 DISCUSSION

In this study I compared three types of vector space models for automatically predicting exam grades. The vector space models that used topics (LDA) or singular values (LSA) as their dimensions both outperformed the baseline vector space models that used vocabulary counts as their dimensions. The likely explanation for this is that LDA and LSA models compare documents at the semantic level, rather than looking for exact vocabulary matches [6]. Notably, even in the baseline models the correlation between predicted and lecturer-assigned grades was statistically significant.

Looking back on the literature, Kakkonen et al. (2008) asked whether their LSA models perhaps outperformed their LDA models because they were trained on a small set of student answers only. My results support this explanation: when training on the student answers LSA outperformed LDA, but it was the other way around when using a more diverse document collection (i.e., sentences in a psychology textbook chapter). This raises the question as to why the two types of training data affected the topic models differently (the size of the training sets was similar).

One explanation can be sought in the optimal number of topics for the LDA models that resulted from a grid search (see 2.3.3). This number was different for the two types of training data. In the student answers, the grid search had shown that the optimal number of topics was 2. This is the lowest possible number of topics that an LDA model can have. This outcome makes sense because in essence the student answers all revolved around the same topic (i.e., Whorf’s language theory). As a result, however, the resulting vector space models only had two dimensions, which does not allow very fine distinctions between answers of different qualities. When visually inspecting the scatter plots of the outcomes of these models (not printed here), it can be seen that these models predict a 10 for almost all students, resulting in very low correlations. Thus, training on a more diverse corpus such as a textbook seems more suitable when working with LDA models. Indeed, Table 1 shows better results for LDA models that were trained on the textbook and used 20 topics, as compared to LDA models that were trained on student answers.

The textbook-trained LDA models also outperformed the textbook-trained LSA models by a large margin. This is the outcome that Kakkonen et al. (2008) had predicted in their own study. They list several arguments as to why LDA models are expected to function better, including the difficulty to select the right number of dimensions for LSA, the fact that no probability distribution is defined in LSA, and that the reduced matrix in LSA can contain negative values. These explanations likely also apply to the current study. Incidentally, it is nice to see the LDA model achieving the best results, because this is the model whose outcomes are easiest to interpret by students and lecturers.

Zooming in on the individual models, Figures 2-4 show that none of the models ever predicted the grade of 10, even though this was the most common grade in the training set (see Figure 1). Thus, while the lecturer considered many of the students’ answers to be excellent, these answers apparently were quite different from the reference answer in terms of the vector space models that I used for calculating semantic similarity. This shows that Wolfe et al.’s (1998) [2] reference answer approach may not be ideal for grading student answers by means of vector space models.

An alternative approach is described in [9], which may provide a better solution for future research. Rather than comparing a student’s answer to one specific reference answer, the to-be-graded answer is compared to a set of answers already graded by the lecturer. Then, you take the (for example) ten graded answers with the highest similarity score to the to-be-graded answer. The predicted grade will be the weighted average of the grades of the ten most similar answers (weighted by their similarity score).

Another limitation of the current approach (and the alternative approach from [9] as well) is that creative answers are punished, such as when a student uses a unique example. This makes the student’s answer more dissimilar from the reference answer (or from other students’ answers), and results in a lower grade. One solution could be to create a list with terms that are relevant to the question (e.g., [‘Whorf’, ‘linguistic’, ‘relativity’]) and first reduce the student’s answer to only the sentences that contain one of these terms. Sentences such as ‘The Inuit have many words for snow’ would no longer be part of the semantic similarity calculation, and perhaps wrongfully lower the predicted grade.

With regard to the three other variables, spelling correction and the mapping algorithm did not seem to have much of an effect in the baseline and LSA models. In the LDA models they did, but these models are the least reliable for evaluating these variables, because LDA models are probabilistic and their results were seen to fluctuate quite a bit when the same model was run twice. The baseline model worked best with raw counts; for LDA not much of a difference is seen, and for LSA it seems best to use TF-IDF counting, although this cannot be said with much certainty either because of the issue described in Footnote 1.

The evaluation metric in this study was the correlation between the assigned and predicted grades. It would be interesting to consider other metrics too, for example the sum of squared errors (SSE) between the assigned and predicted grades. This would enable us to also evaluate a baseline model that would assign the most common grade (in this case 10) to all student answers. Such a baseline model currently could not be employed because a correlation coefficient cannot be calculated when there is no variance in the data.

The important question to close with is of course whether the models in this study could have any real-life implications. While it was good to see that all of the models yielded a statistically significant correlation between predicted and assigned grades, it still seems too early to justify their actual implementation in higher education. We would need to know what is the correlation between two human graders, and equal or go beyond that correlation with our model. Furthermore, the model needs to be able to assign the maximum grade of 10 to excellent answers that are different from the reference answer. With the suggestions proposed in this discussion, we should continue to try to find a reliable method for automatizing the grading process. This could be of great benefit to students, as they could be provided with feedback on their learning process much more quickly.

REFERENCES

[1] Page, E. B. (1966). Grading essays by computer: Progress report. *Notes from the 1966 Invitational Conference on Testing Problems*, 87–100.

[2] Wolfe, M. B., Schreiner, M., Rehder, B., Laham, D., Foltz, P. W., Kintsch, W., & Landauer, T. K. (1998). Learning from text: Matching readers and texts by Latent Semantic Analysis. *Discourse Processes*, 25(2–3), 309–336. http://doi.org/10.1080/01638539809545030

[3] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3, 993–1022. http://doi.org/10.1162/jmlr.2003.3.4-5.993

[4] Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The Latent Semantic Analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2), 211–240.

[5] Alves dos Santos, J. C., & Favero, E. L. (2015). Practical use of a latent semantic analysis (LSA) model for automatic evaluation of written answers. *Journal of the Brazilian Computer Society*, 21(21), 1–8. http://doi.org/10.1186/s13173-015-0039-7

[6] Kakkonen, T., Myller, N., Sutinen, E., & Timonen, J. (2008). Comparison of dimension reduction methods for automated essay grading. *Educational Technology & Society*, 11(3), 275–288.

[7] Gleitman, H., Gross, J., & Reisberg, D. (2011). *Psychology* (8th ed.). New York, NY: W. W. Norton & Company.

[8] Bird, S., Loper, E., & Klein, E. (2009). *Natural language processing with Python*. Sebastopol, CA: O’ReillyMedia Inc.

[9] Foltz, P. W., Laham, D., & Landauer, T. K. (1998). The Intelligent Essay Assessor: Applications to educational technology. *Interactive Multimedia Journal of Computer-Enhanced Learning*, 1–9.