Multiple Instance Learning for Content Feedback Localization without Annotation

Scott Hellman, William R. Murray, Adam Wiemerslage, Mark Rosenstein,
Peter W. Foltz, Lee Becker and Marcia Derr
Pearson
Boulder, CO
{scott.hellman, william.murray, adam.wiemerslage, mark.rosenstein, peter.foltz,
lee.becker, marcia.derr}@pearson.com

Abstract

Automated Essay Scoring (AES) can be used to automatically generate holistic scores with reliability comparable to human scoring. In addition, AES systems can provide formative feedback to learners, typically at the essay level. In contrast, we are interested in providing feedback specialized to the content of the essay, and specifically for the content areas required by the rubric. A key objective is that the feedback should be localized alongside the relevant essay text. An important step in this process is determining where in the essay the rubric designated points and topics are discussed. A natural approach to this task is to train a classifier using manually annotated data; however, collecting such data is extremely resource intensive. Instead, we propose a method to predict these annotation spans without requiring any labeled annotation data. Our approach is to consider AES as a Multiple Instance Learning (MIL) task. We show that such models can both predict content scores and localize content by leveraging their sentence-level score predictions. This capability arises despite never having access to annotation training data. Implications are discussed for improving formative feedback and explainable AES models.

1 Introduction

The assessment of writing is an integral component in the pedagogical use of constructed response items. Often, a student's response is scored according to a rubric that specifies the components of writing to be assessed – such as content, grammar, and organization – and establishes an ordinal scale to assign a score for each of those components. Furthermore, there is strong evidence of learning improvements when instructors provide feedback to their students (Graham et al., 2011). Their comments can take the form of holistic, document-level feedback, or more specific, targeted feedback that addresses an error or praises an insight at relevant locations in the paper.

As far back as the 1960s, computers have been employed in essay scoring (Page, 1966). Thus, automated essay scoring (AES) is a well-studied area, and with modern approaches, AES systems are often as reliable as human scorers (Shermis and Burstein, 2003, 2013). However, many of these systems are limited to providing holistic scores – that is, they assign an ordinal value for every component in the rubric.

Furthermore, some AES systems can provide document-level feedback, but this requires students to interpret which parts of their text the feedback refers to. When an automated scoring system additionally provides location information, students can leverage a more specific frame of reference to better understand the feedback. Indeed, students are more likely to understand and implement revisions when given feedback that summarizes and localizes relevant information (Patchan et al., 2016).

We are interested in automatically providing localized feedback on the content of an essay. The specific kinds of feedback provided can vary, ranging from positive feedback reinforcing that a student correctly covered a specific topic, to feedback indicating areas that the student could improve. This latter category includes errors such as domain misconceptions or inadequate citations. We consider wholly omitted topics to be outside the scope of localized feedback, as they represent an overall issue in the essay that is best addressed by essay-level feedback.

From a machine learning perspective, content localization is difficult. Current automated localization is often very fine-grained, e.g., grammar checkers can identify spelling or grammar mistakes at the word level. However, we view the content of a student's essay as primarily a sentence-level aspect of student writing. Critically, to provide this type of content feedback, we need to be able to

Proceedings of the 15th Workshop on Innovative Use of NLP for Building Educational Applications, pages 30–40
July 10, 2020. ©2020 Association for Computational Linguistics

detect where in their essay a student is discussing that particular content. One approach would be to collect a corpus of training data containing essays with annotations indicating text spans where topics of interest were discussed. A supervised machine learning classifier could be trained on this data, and this localization model could then be integrated into a full AES feedback system. For example, a scoring model could identify the degree of coverage of rubric-required topics $t_1, \ldots, t_n$. A formative feedback system could generate suggestions for inadequately covered topics. Finally, the localization system could identify *where* this formative feedback should be presented. In this work, we address the localization part of this process.

While AES systems typically provide scoring of several rubric traits, we are interested primarily in the details of an essay's content, and so our work here focuses on a detailed breakdown of content coverage into individual topics. For example, consider a prompt that asks students to discuss how to construct a scientific study on the benefits of aromatherapy. Each student answer is a short essay, and is scored on its coverage of six content topics. Examples of these topics include discussion of independent and dependent variables, defining a blind study, and discussing the difficulties in designing a blind study for aromatherapy. These kinds of content topics are what our localization efforts are focused on. Figure 1 shows a a screenshot from an annotation tool containing an example essay with human-provided annotations and scores.

The downside of building a localization classifier based on annotation data is that such annotation data is very expensive to collect. Holistic scoring data itself is expensive to collect, and obtaining reliable annotations is even more difficult to orchestrate. Due to these issues, an approach that eliminates annotation training data is desirable. We propose a weakly-supervised multiple instance learning (MIL) approach to content localization, that relies on either document-level scoring information, or on a set of manually curated reference sentences. We show that both approaches can perform well at the topic localization task, without having been trained on localization data.

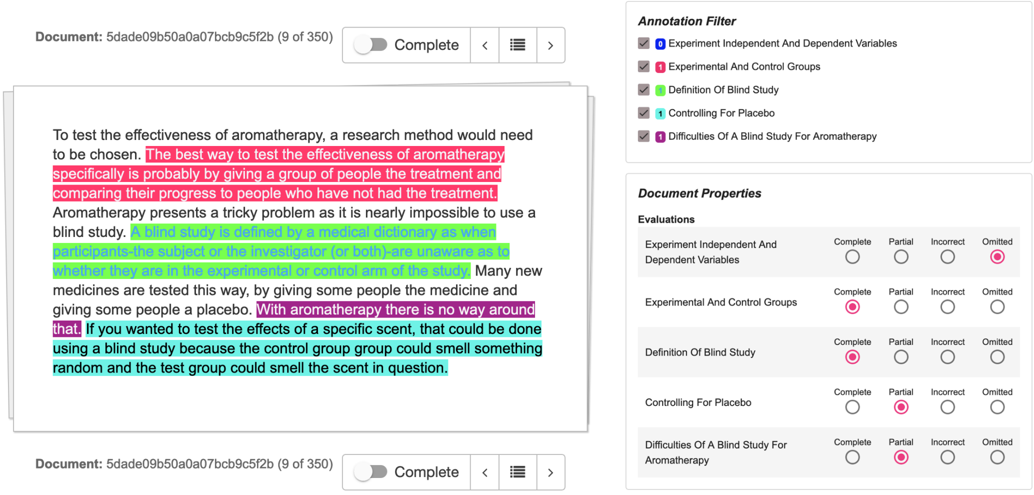
2 Automated Essay Scoring and Feedback

Automated Essay Scoring systems for providing holistic scoring are well studied (Shermis and Burstein, 2003, 2013). Some systems are specifically designed to provide formative feedback, with or without an accompanying overall score. Roscoe et al. (2012) presents an automated feedback system that measures attributes of the student response and provides specific feedback if certain thresholds are met (e.g., "use larger words" when the mean syllables per word is too low). In Foltz et al. (2000) an AES system is shown that uses Latent Semantic Analysis (LSA) to measure similarities between student sentences and reference sentences. Each required topic has a set of 1–3 reference sentences, and if no sentence in the student essay is similar to any reference sentences for that topic, feedback encouraging the student to more fully describe the topic is presented. Summary Street® provides students with content feedback during the summarization task, and specifically uses a reference document with LSA for semantic comparison (Steinhart, 2001; Franzke et al., 2005).

There has been effort toward providing students with localized feedback as well. Burstein et al. (2003) presents a system that uses an ensemble of supervised machine learning models to locate and provide feedback on discourse components such as thesis statements. Similarly, Chukharev-Hudilainen and Saricaoglu (2016) presents a system that provides feedback on discourse structure in essays written by English language learners.

A major drawback of these more localized feedback systems is the requirement that they be trained on annotation data, which is expensive to gather. Our work, which removes this constraint, is inspired by approaches that determine the contribution of individual sentences to the overall essay score. One such approach is described in Dong et al. (2017), which presents a neural network that generates an attention vector over the sentences in a response. This attention vector directly relates to the importance of each individual sentence in the computation of the final predicted score.

Woods et al. (2017) attempts to localize feedback based purely on the output of a holistic AES model. Specifically, they train an ordinal logistic regression model on a feature space consisting of character, word, and part-of-speech n-grams. They show that this model performs well on the AES task. They then propose a method for determining the contribution of each sentence to the overall score by measuring how much more likely a lower (or higher) score would be if that sentence was re-



Document: 5dade09b50a0a07bcb9c5f2b (9 of 350)

To test the effectiveness of aromatherapy, a research method would need to be chosen. The best way to test the effectiveness of aromatherapy specifically is probably by giving a group of people the treatment and comparing their progress to people who have not had the treatment. Aromatherapy presents a tricky problem as it is nearly impossible to use a blind study. A blind study is defined by a medical dictionary as when participants-the subject or the investigator (or both)-are unaware as to whether they are in the experimental or control arm of the study. Many new medicines are tested this way, by giving some people the medicine and giving some people a placebo. With aromatherapy there is no way around that. If you wanted to test the effects of a specific scent, that could be done using a blind study because the control group group could smell something random and the test group could smell the scent in question.

Document: 5dade09b50a0a07bcb9c5f2b (9 of 350)

Annotation Filter

- ☑ 0 Experiment Independent And Dependent Variables
- ☑ 1 Experimental And Control Groups
- ☑ 1 Definition Of Blind Study
- ☑ 1 Controlling For Placebo
- ☑ 1 Difficulties Of A Blind Study For Aromatherapy

Document Properties

Evaluations

	Complete	Partial	Incorrect	Omitted
Experiment Independent And Dependent Variables	○	○	○	●
Experimental And Control Groups	●	○	○	○
Definition Of Blind Study	●	○	○	○
Controlling For Placebo	○	●	○	○
Difficulties Of A Blind Study For Aromatherapy	○	●	○	○

Figure 1: Screenshot from an annotation tool containing an example essay with colored text indicating human-provided annotations (left), the color-coded annotation key (top right) and holistic scores (bottom right).

moved. They then use the Mahalanobis distance to compute how much that sentence's contribution differs from a known distribution of sentence contributions. Finally, they present feedback to the student, localized to sentences that were either noticeably beneficial or detrimental to the overall essay.

We are interested in almost exactly the same task as Woods et al. (2017) – the only difference is that we aim to predict the locations humans would annotate, while their goal was to evaluate the effectiveness of their localized feedback. Specifically, we frame annotation prediction as a task with a set of essays and a set of labels, such that each sentence in each essay has a binary label indicating whether or not the specified topic was covered in that sentence. The goal is to develop a model that can predict these binary labels given the essays.

Latent Dirichlet Allocation (LDA) is an unsupervised method for automatically identifying topics in a document (Blei et al., 2003), and is related to our goal of identifying sentences that received human annotations. This requires an assumption that the human annotators identified sentences that could match a specific topic learned by LDA. While there is some work on using LDA to aid in annotation (Camelin et al., 2011), we are unaware of any attempts to extend it to the educational writing domain. Our approach differs from LDA in that we use supervised techniques whose predictions can be transferred to the annotation domain, rather than approaching the problem as a wholly unsupervised task. Additionally, we are classifying sentences by topics rather than explicitly creating word topic models for the topics.

If one views student essays as summaries (e.g., of the section of the textbook that the writing prompt corresponds to), then summarization evaluation approaches could be applicable. In particular, the PEAK algorithm (Yang et al., 2016) builds a hypergraph of subject-predicate-object triples, and then salient nodes in that graph are identified. These salient nodes are then collected into summary content units (SCUs), which can be used to score summaries. In our case, these SCUs would correspond to recurring topics in the student essays. One possible application of PEAK to our annotation prediction problem would be to run PEAK on a collection of high-scoring student essays. Similarity to the identified SCUs could then be used as a weak signal of the presence of a human annotation for a given sentence. Our approach differs from this application of PEAK in that we not only utilize similarity to sentences from high-scoring essays, but also use sentences from low-scoring essays as negative examples for a given topic.

3 Multiple Instance Learning

To accomplish our goal of predicting annotations without having access to annotation data, we approach AES as a multiple instance learning regression problem. Multiple instance learning is a supervised learning paradigm in which the goal is to label bags of items, where the number of items in a bag can vary. The items in a bag are also referred to as *instances*. MIL is a well-studied area of machine learning, with a broad literature into its applications both in NLP (e.g., Bunescu and Mooney (2007)) and in general settings (e.g., Dietterich et al. (1997)). The description provided here is based on Carbonneau et al. (2016).

The standard description of MIL assumes that the goal is a binary classification. Intuitively, each bag has a known binary label, and we can think of the instances in a bag as having unknown binary labels. We then assume that the bag label is some aggregation of the unknown instance labels. We first describe MIL in these terms, and then extend those ideas to regression.

Formally, let $\mathcal{X}$ denote our collection of training data, and let i denote an index over bags, such that each $X_i \in \mathcal{X}$ is of the form $X_i = \{x_{i,1}, x_{i,2}, \ldots, x_{i,m}\}$. Note that m can differ among the elements of X, that is, the cardinalities of two elements $X_i, X_j \in \mathcal{X}$ need not be equal. Let Y denote our training labels, such that each X_i has a corresponding $Y_i \in \{0, 1\}$. We assume that there is a latent label for each instance $x_{i,j}$, denoted by $y_{i,j}$. Note that, in our specific application, $x_{i,j}$ corresponds to the j-th sentence of the i-th document in our corpus. The *standard assumption* in MIL asserts that

$$Y_i = \begin{cases} 0 & \text{if } \forall x_{i,j} \in X_i, y_{i,j} = 0 \\ 1 & \text{if } \exists x_{i,j} \in X_i, y_{i,j} = 1 \end{cases}$$

That is, the standard assumption holds that a bag is positive if *any* of its constituent instances are positive. Another way of framing this assumption is that a single instance is responsible for an entire bag being positive.

In contrast, the *collective assumption* holds that Y_i is determined by some aggregation function over *all* of the instances in a bag. Thus, under the collective assumption, a bag's label is dependent upon more than one and possibly all of the instances in that bag.

AES is usually approached as a regression task, so these notions must be extended to regression. We adapt the standard assumption, that a single instance determines the bag label, by using a function that selects a single instance value from the bag. In this work, we use the maximum instance label. We adapt the collective assumption, that all instance labels contribute to the bag label, by using a function that aggregates across all instance labels. In this work, we use the mean instance label.

The application of MIL to natural language processing tasks is quite common. Wang et al. (2016) trains a convolutional neural network to aggregate predictions across sentences in order to predict discussion of events in written articles. By framing this task as a MIL problem, not only can they learn to predict the types of events articles pertain to, they can also predict which sentences specifically discuss those events. A variety of similar approaches that assign values to sentences and then use aggregation to create document scores have been used for sentiment analysis (Kotzias et al., 2015; Pappas and Popescu-Belis, 2017; Angelidis and Lapata, 2018; Lutz et al., 2019).

To the best of our knowledge, applications of MIL in educational domains are rare, and we are not aware of any attempts to explicitly approach AES as a MIL task. The educational MIL work that we are aware of uses MIL to determine overall student performance given their trajectory over the duration of a course (Zafra et al., 2011).

4 Automated Essay Scoring with Multiple Instance Learning

By framing AES as a MIL problem, the goal becomes predicting, for each sentence, the score for that sentence, and then aggregating those sentence-level predictions to create a document-level prediction. This goal requires determining both how to predict these sentence-level scores, and how to aggregate them into document-level scores. Note that we perform this task independently for each topic $t_1, \ldots, t_n$, but this discussion is limited to a single topic for clarity.

We define the AES task as follows. Assume we are given a collection of student essays D and corresponding scores y. We assume these scores are numeric and lie in a range defined by the rubric – we use integers, but continuous values could also work. For example, if the rubric for a concept defined the possible scores as *Omitted/Incorrect*, *Partially Correct*, and *Correct*, the corresponding entries in y could be drawn from $\{0, 1, 2\}$. The AES task is to predict y given D.

The intuition for why MIL is appropriate for AES is that, for many kinds of topics, the content of a single sentence is sufficient to determine a score. For example, consider a psychology writing prompt that requires students to include the definition of a specific kind of therapy. If an essay includes a sentence that correctly defines that type of therapy, then the essay as a whole will receive a high score for that topic.

We approach the sentence-level scoring task using k-Nearest Neighbors (kNN) (Cover and Hart, 1967). Denote the class label of a training example

a as y_a. For each document in our training corpus, we project each sentence into a semantic vector space, generating a corresponding vector that we denote as x. We assign to x the score of its parent document. We then train a kNN model on all of the sentences in the training corpus. We use the Euclidean distance as the metric for our nearest neighbor computations.

To predict the score of a new document using this model, we first split the document into sentences, project those sentences into our vector space, and use the kNN model to predict the score of each sentence. We define this sentence-level scoring function ϕ as

$$\phi(x) = \frac{1}{k} \sum_{a \in \text{knn}(x)} y_a$$

where $\text{knn}(x)$ denotes the set of k nearest neighbors of x. We aggregate these sentence-level scores through a document-level scoring function θ:

$$\theta(X_i) = \underset{x_{i,j} \in X_i}{\text{agg}} (\phi(x_{i,j}))$$

where agg corresponds to either the maximum or the mean – that is, agg determines whether we are making the standard or collective assumption.

We consider three semantic vector spaces. We define our vocabulary V as the set of all words appearing in the training sentences. The first vector space is a tf-idf space, in which each sentence is projected into $\mathbb{R}^{|V|}$ and each dimension in that vector corresponds to the term frequency of the corresponding vocabulary term multiplied by the inverse of the number of documents that contained that term.

We also consider a pretrained latent semantic analysis space. This space is constructed by using the singular value decomposition of the tf-idf matrix of a pretraining corpus to create a more compact representation of that tf-idf matrix (Landauer et al., 1998).

Finally, we consider embedding our sentences using SBERT (Reimers and Gurevych, 2019). SBERT is a version of BERT (Devlin et al., 2019) that has been fine-tuned on the SNLI (Bowman et al., 2015) and Multi-Genre NLI (Williams et al., 2018) tasks. These tasks involves predicting how sentences relate to one another. Critically, this means that the SBERT network has been specifically fine-tuned to embed individual sentences into a common space.

5 Weakly Supervised Localization

While this kNN-MIL model is ultimately trained to predict document-level scores for essays, as a side effect, it also generates a score prediction for each sentence. The central idea is that we can directly use these sentence-level scores as weak signals of the presence of annotation spans in the sentences.

Concretely, given our trained kNN-MIL model and an essay X_i, we predict the presence of annotations as follows. Assume that the minimum and maximum scores allowed by the rubric for the given topic are S_{min} and S_{max}, respectively. We leverage the sentence-level scoring function ϕ to compute an annotation prediction function α:

$$\alpha(x_{i,j}) = \frac{\phi(x_{i,j}) - S_{min}}{S_{max} - S_{min}}$$

That is, our annotation prediction function α is a rescaling of ϕ such that it lies in $[0, 1]$, allowing us to interpret it as a normalized prediction of a sentence having an annotation.

As our goal is to predict annotation spans without explicit annotation data, we also consider a modification of this process. Rather than training our kNN-MIL model on a corpus of scored student essays, we could instead use a set of manually curated reference sentences to train the model. We consider two sources of reference sentences.

First, we consider reference sentences pulled from the corresponding rubric, labeled by the topic they belong to. Rubrics often have descriptions of ideal answers and their key points, so generating such a set is low-cost. However, sentences from rubric descriptions may not discuss a topic in the same way that a student would, or they may fail to anticipate specific correct student answers.

For these reasons, we also consider selecting reference sentences by manually picking sentences from the training essays. We consider all training essays that received the highest score on a topic as candidates and choose one to a few sentences that clearly address the topic. We specifically look for exemplars making different points and written in different ways. These identified sentences are manually labeled as belonging to the given topic, and each one is used as a different reference sentence when training our kNN-MIL model. Typically, just a few exemplars per topic is sufficient (Foltz et al., 2000).

Whether we collect examples of formal wording from the rubric or informal wording from student answers, or both, we must then label the reference sentences for use in our kNN-MIL model. For a given topic, the references drawn from other topics provide negative examples of it. To convert these manual binary topic labels into the integer space that we use for the AES task, we assign to each reference sentence the maximum score for the topic(s) it was labeled as belonging to, and the minimum score to it for all other topics.

The key benefit of our approach is that it never requires access to annotation training data. Instead, given a collection of student essays for a new prompt, training a kNN-MIL model for that prompt requires one of a few sources of data. If we have human-provided document-level scores for the topics we are interested in, we can train a kNN-MIL model on those labeled documents. Otherwise, if the rubric contains detailed enough reference sentences and descriptions for the various topics, we can train a kNN-MIL model using reference sentences collected from the rubric. And finally, we can have a human expert collect examples of the topics of interest from the essays, and then train a kNN-MIL model using those examples as reference sentences.

6 Datasets

To evaluate the performance of kNN-MIL, we need student essays that have both document-level scores and annotation spans. To the best of our knowledge, there is no publicly available dataset that contains both.

Thus, we make use of an existing Pearson proprietary corpus developed to explore fine-grained content assessment for formative feedback. This corpus consists of student responses to four university-level psychology writing prompts. While the essays were originally written and scored against holistic writing traits, a subsequent annotation effort factored the content trait into multiple topics that represent core ideas or assertions an instructor would expect a student to address within the essay. For example, the topic *Comparing Egocentrism* from a prompt about Piaget's stages of development has the following reference answer:

> *A child in the pre-operational stage is unable to see things from another person's point of view, whereas a child in the concrete operational stage can.*

Annotators were tasked with assigning an essay-level rating for each topic with a judgment of *Complete*, *Partial*, *Incorrect* or *Omitted*. Additionally, they were asked to mark spans in the essay pertaining to the topic – these could be as short as a few words or as long as multiple sentences. Two psychology subject matter experts (SMEs) performed the rating and span selection tasks. Ideally, rating and span annotations would have also been adjudicated by a third SME. However, due to time and cost constraints, we lack adjudicated labels for three of the four prompts. For this reason, we ran our experiments on both annotators separately.

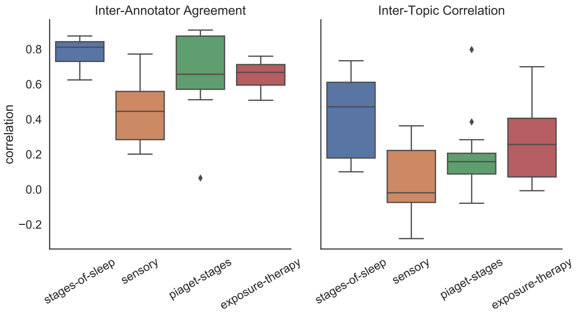


Figure 2: Box plots of inter-annotator correlations of the sentence-level annotation labels for each topic (left) and correlation between scores for all topic pairs (right).

As our techniques work at a sentence-level, but the human annotations can be shorter or longer than a single sentence, we frame the annotation prediction task as the task of predicting, for a given sentence, whether an annotation overlapped with that sentence. We show the distribution of inter-annotator agreements for the topics in the four prompts in the left panel of Figure 2, calculated as the correlation between these sentence-level annotation labels. The annotators achieved reasonable reliability except on the Sensory prompt, where the median correlation was below 0.5, and one topic in the Piaget prompt, where the annotators had a correlation near 0.

The features of these four prompts are shown in Table 1. Essays had 5–8 topics and covered areas such as the stages of sleep; the construction of a potential experimental study on aromatherapy; Piaget's stages of cognitive development; and graduated versus flooding approaches to exposure therapy for a hypothetical case of agoraphobia. Table 2 shows how many sentences were available for training the kNN-MIL models for each prompt.

Our approach assumes that the topic scores are numeric. We convert the scores in this dataset by mapping both *Omitted* and *Incorrect* to 0, *Partial*

Prompt	# of Essays	# of Topics	Mean Words	Annotator 1	Annotator 2
Sleep Stages	283	7	361	9%	8%
Sensory Study	348	6	395	7%	14%
Piaget Stages	448	8	367	10%	6%
Exposure Therapy	258	5	450	15%	9%

Table 1: Characteristics and summary statistics of prompts used in the experiments. The Annotator columns indicate, for a specific topic, the average percentage of sentences annotated with that topic.

Prompt	Rubric	Student	Training
Sleep Stages	15	19	4741
Sensory Study	11	13	5362
Piaget Stages	26	22	6342
Exposure Therapy	20	48	5184

Table 2: Number of sentences available for kNN-MIL training. The Rubric column shows the number of reference sentences taken from the rubric, while the Student column shows the number manually chosen from the student essays. The Training column shows the total number of sentences in the full set of essays.

to 1, and *Complete* to 2. As our approach uses these topic scores to generate annotation predictions, its ability to predict different annotations for different topics depends on the topic scores not being highly correlated. The right panel of Figure 2 shows the distribution of inter-topic correlations for each prompt. While there is considerable variation between the prompts, we do see that, except for one topic pair on the Piaget prompt, all inter-topic correlations are less than 0.8, and the median correlations are all below 0.5.

7 Experiments

Our goal is to determine how well the kNN-MIL approaches perform on the annotation prediction task. We also want to verify that our approaches perform reasonably well on the essay scoring task – while we are not directly interested in essay scoring, if our approaches are incapable of predicting essay scores, that would indicate that the underlying assumptions of our kNN-MIL approaches are likely invalid.

For each prompt, we construct 30 randomized train/test splits, holding out 20% of the data as the test set. We then train and evaluate our models on those splits, recording two key values: the correlation of the model's document-level scores to the human scorer, and the area under the ROC curve of the model's sentence-level annotation predictions.

We compare results between three categories of models. The first is the kNN-MIL model, trained on the training set. We refer to this model as the Base kNN-MIL model. The second is the kNN-MIL model trained on a manually curated reference set, which we refer to as the Manual kNN-MIL model. Finally, we compare to the ordinal logistic regression-based approach presented in Woods et al. (2017), which we will refer to as the OLR model. Additionally, as a baseline for comparison on the annotation prediction task, we train a sentence-level kNN model directly on the human annotation data, which we refer to as the Annotation kNN model. We consider the Annotation kNN model to provide a rough upper bound on how well the kNN-MIL approaches can perform. Finally, for our kNN-MIL models, we investigate how varying k and the vector space impacts model performance.

We use the all-threshold ordinal logistic regression model from mord (Pedregosa-Izquierdo, 2015) and the part of speech tagger from spaCy (Honnibal and Montani, 2017) in our implementation of the OLR model. The Mahalanobis distance computation for this approach requires a known distribution of score changes, for this we use the distribution of score changes of the training set.

We use the kNN and tf-idf implementations from scikit-learn (Pedregosa et al., 2011) and the LSA implementation from gensim (Řehůřek and Sojka, 2010). Our pretrained LSA space is 300 dimensional, and is trained on a collection of 45,108 English documents sampled from grade 3-12 readings and augmented with material from psychology textbooks. (Landauer et al., 1998). After filtering very common and uncommon words, this space includes 37,013 terms, covering 85% of the terms appearing in the training data.

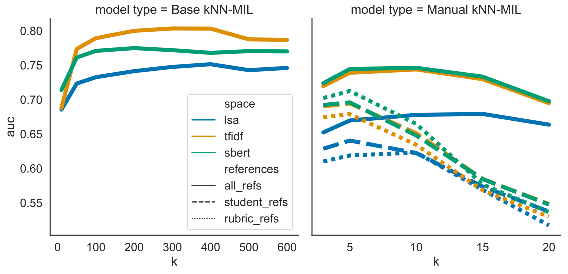


Figure 3: Annotation prediction performance of the kNN-MIL models as k is varied, averaged across all prompts, concepts, and annotators. Error bars omitted for clarity.

8 Discussion

We present the average annotation prediction performance of the kNN-MIL models for different values of k in Figure 3. While all approaches achieve AUCs above 0.5, the LSA-based space performs relatively poorly. The tf-idf space performs well, especially for the Base kNN-MIL model. In the tf-idf space, Base kNN-MIL performance peaks at $k = 400$. For the Manual kNN-MIL models, best performance occurs with the combined reference set using the tf-idf or SBERT spaces, around $k = 10$. Performance for Manual kNN-MIL with only rubric references or student references peaks and declines sooner than for combined due to the set of possible neighbors being smaller.

Note that the substantial difference in k between Base kNN-MIL and Manual kNN-MIL is due to the fact that we have orders of magnitude fewer manual reference sentences than training set sentences.

In light of these results, for clarity in the rest of this discussion, we focus on $k = 400$ for Base kNN-MIL, $k = 10$ and the combined reference set for Manual kNN-MIL, and exclude the LSA space.

To determine how annotation prediction differs across model types, we show the average overall AUC of all models in Table 3. In this table, we see that our best performance is achieved when we train a kNN model on actual annotation data. In contrast, the OLR model performs relatively poorly, suggesting that its success at predicting sentences that require some sort of feedback does not directly translate into an ability to predict locations of annotations.

Between the different kNN-MIL approaches, Base kNN-MIL using a tf-idf vector space performs best on three of the four prompts, and regardless of vector space, Base kNN-MIL performs as well or better than Manual kNN-MIL on those same three prompts. On the remaining prompt, Exposure Therapy, Manual kNN-MIL with SBERT performs best, but the differences between the various kNN-MIL approaches are relatively small on this prompt.

These annotation predictions results show that the kNN-MIL approach performs well despite never being explicitly trained on the annotation prediction task. While the Base kNN-MIL approach is overall better than the Manual kNN-MIL approach, it also requires a large amount of scored data for training. Which kNN-MIL approach is best for a particular situation thus depends on if the additional performance gain of Base kNN-MIL is worth the added cost of obtaining essay scoring data.

Finally, we show performance on the essay scoring task in Table 4. On this task, the OLR model and the Base kNN-MIL model with a tf-idf space perform the best, and the Manual kNN-MIL models perform the worst. We had predicted that the standard MIL assumption would perform well for AES, and our results show that this is true – for both Base and Manual kNN-MIL, using the maximum sentence topic score in an answer outperforms using the mean sentence topic score.

The Base kNN-MIL model can perform relatively well at both the document scoring task and the annotation prediction task. This suggests that it could be used as an explainable AES model, as the annotation predictions are directly tied to the document-level scores it provides. In this quite different application, the localization would be used to explain the sentences contributing to the final score, rather than to provide context for formative feedback.

9 Conclusions and Future Work

We have presented a novel approach of using MIL to train annotation prediction models without access to annotation training data. This technique performs well and can allow for automated localization without expensive data annotation. It also performs relatively well on the document-level scoring task, suggesting that its sentence-level score predictions could be used as part of an explainable model for AES.

Given that our kNN-MIL approach operates at the sentence level, it is unlikely to correctly locate annotations that exist across multiple sentences. Adapting our method to better incorporate information across sentences (e.g., by incorporating co-

Model	Space	Exposure Therapy	Piaget Stages	Sensory Study	Sleep Stages
Annotation kNN	sbert	0.88 (0.04)	0.89 (0.08)	0.85 (0.06)	0.91 (0.03)
	tfidf	0.87 (0.04)	0.92 (0.07)	0.89 (0.06)	0.93 (0.02)
Base kNN-MIL	sbert	0.76 (0.08)	0.78 (0.09)	0.77 (0.09)	0.78 (0.06)
	tfidf	0.74 (0.06)	0.84 (0.10)	0.81 (0.09)	0.80 (0.07)
Manual kNN-MIL	sbert	0.78 (0.07)	0.73 (0.12)	0.70 (0.10)	0.78 (0.06)
	tfidf	0.74 (0.08)	0.77 (0.09)	0.68 (0.10)	0.75 (0.07)
OLR		0.55 (0.04)	0.63 (0.08)	0.63 (0.07)	0.61 (0.05)

Table 3: Area under the ROC curve on the annotation prediction task, averaged over all topics and annotators. Standard deviation shown in parentheses.

Model	agg	Space	Exposure Therapy	Piaget Stages	Sensory Study	Sleep Stages
Base kNN-MIL	max	sbert	0.49 (0.14)	0.51 (0.18)	0.41 (0.15)	0.60 (0.11)
		tfidf	0.47 (0.12)	0.61 (0.19)	0.52 (0.17)	0.67 (0.12)
	mean	sbert	0.39 (0.15)	0.44 (0.16)	0.36 (0.15)	0.61 (0.14)
		tfidf	0.40 (0.14)	0.52 (0.16)	0.46 (0.14)	0.63 (0.13)
Manual kNN-MIL	max	sbert	0.41 (0.15)	0.30 (0.18)	0.25 (0.15)	0.37 (0.14)
		tfidf	0.38 (0.14)	0.40 (0.15)	0.23 (0.16)	0.34 (0.18)
	mean	sbert	0.29 (0.15)	0.23 (0.15)	0.16 (0.15)	0.27 (0.14)
		tfidf	0.29 (0.16)	0.29 (0.13)	0.19 (0.16)	0.22 (0.20)
OLR			0.50 (0.18)	0.63 (0.16)	0.51 (0.18)	0.69 (0.14)

Table 4: Pearson correlation coefficients on the document-level scoring task, averaged over all topics. Standard deviation shown in parentheses.

reference resolution) could help improve its overall performance. Additionally, as the Base kNN-MIL approach uses topics as negative examples for each other, we expect that it would not work well in situations where the inter-topic score correlations were high. We expect the Manual kNN-MIL approach to be less sensitive to this issue. Determining other ways to include negative examples would allow the Base kNN-MIL approach to be applied to prompts whose topics were highly correlated.

In our current domain, psychology, and in the context of low-stakes formative feedback, incorrect answers are uncommon compared to omitted or partial answers. In contrast, for domains that require chained reasoning over more complex mental models, such as accounting, cell biology, or computer science, we expect the ability to correctly detect misconceptions and errors to be far more important. In general, future work is required to determine how well our approach will work in other domains, and which domains it is best suited to.

Determining where topics are discussed is only one step in the full formative feedback process. More work is required to determine the path from holistic scoring and topic localization to the most helpful kinds of feedback for a student. In particular, we need to consider different kinds of pedagogical feedback and how such feedback could be individualized. Additionally, we could provide not just text but also video, peer interaction, worked examples, and other approaches from the full panoply of potential pedagogical interventions. Finally, we need to decide what actions will help the student the most, which relies on our pedagogical theory of how to help a student achieve their current instructional objectives.

Acknowledgements

We would like to thank Alok Baikadi, Julio Bradford, Jill Budden, Amy Burkhardt, Dave Farnham, Andrew Gorman and Jorge Roccatagliata for their efforts in collecting the annotated dataset used in this work.

References

Stefanos Angelidis and Mirella Lapata. 2018. Multiple instance learning networks for Fine-Grained sentiment analysis. *Transactions of the Association for Computational Linguistics*, 6:17–31.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research.*, 3(Jan):993–1022.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Razvan Bunescu and Raymond Mooney. 2007. Learning to extract relations from the web using minimal supervision. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 576–583.

Jill Burstein, Daniel Marcu, and Kevin Knight. 2003. Finding the write stuff: Automatic identification of discourse structure in student essays. *Intelligent Systems, IEEE*, 18:32 – 39.

Nathalie Camelin, Boris Detienne, Stéphane Huet, Dominique Quadri, and Fabrice Lefèvre. 2011. Unsupervised concept annotation using latent dirichlet allocation and segmental methods. In *Proceedings of the First workshop on Unsupervised Learning in NLP*, pages 72–81.

Marc-André Carbonneau, Veronika Cheplygina, Eric Granger, and Ghyslain Gagnon. 2016. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition*.

Evgeny Chukharev-Hudilainen and Aysel Saricaoglu. 2016. Causal discourse analyzer: improving automated feedback on academic ESL writing. *Computer Assisted Language Learning*, 29(3):494–516.

T. Cover and P. Hart. 1967. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. 1997. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31–71.

Fei Dong, Yue Zhang, and Jie Yang. 2017. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 153–162.

Peter W Foltz, Sara Gilliam, and Scott A Kendall. 2000. Supporting Content-Based feedback in On-Line writing evaluation with LSA. *Interactive Learning Environments*, 8(2):111–127.

Marita Franzke, Eileen Kintsch, Donna Caccamise, Nina Johnson, and Scott Dooley. 2005. Summary street®: Computer support for comprehension and writing. *Journal of Educational Computing Research*, 33:53–80.

Steve Graham, Karen Harris, and Michael Hebert. 2011. Informing Writing: The Benefits of Formative Assessment. A Report from Carnegie Corporation of New York. *Carnegie Corporation of New York*.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Dimitrios Kotzias, Misha Denil, Nando de Freitas, and Padhraic Smyth. 2015. From group to individual labels using deep features. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 597–606. ACM.

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

Bernhard Lutz, Nicolas Pröllochs, and Dirk Neumann. 2019. Sentence-level sentiment analysis of financial news using distributed text representations and multi-instance learning. In *Proceedings of the 52nd Hawaii International Conference on System Sciences*.

Ellis B Page. 1966. The imminence of grading essays by computer. *The Phi Delta Kappan*, 47(5):238–243.

Nikolaos Pappas and Andrei Popescu-Belis. 2017. Explicit document modeling through weighted multiple-instance learning. *Journal of Artificial Intelligenece Research.*, 58(1):591–626.

Melissa Patchan, Christian Schunn, and Richard Correnti. 2016. The nature of feedback: How peer feedback features affect students' implementation rate and quality of revisions. *Journal of Educational Psychology*, 108.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Fabian Pedregosa-Izquierdo. 2015. *Feature extraction and supervised learning on fMRI : from practice to theory*. Theses, Université Pierre et Marie Curie - Paris VI.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Rod Roscoe, Danica Kugler, Scott A Crossley, Jennifer L Weston, and Danielle McNamara. 2012. Developing pedagogically-guided threshold algorithms for intelligent automated essay feedback. In *Proceedings of the 25th International Florida Artificial Intelligence Research Society Conference, FLAIRS-25*, pages 466–471.

Mark D. Shermis and Jill C. Burstein, editors. 2003. *Automated essay scoring: A cross-disciplinary perspective*. Lawrence Erlbaum Associates, Inc., Mahway, NJ.

Mark D. Shermis and Jill C. Burstein, editors. 2013. *Handbook of automated essay evaluation: Current applications and new directions*. Routledge, New York.

David J. Steinhart. 2001. *Summary street: An intelligent tutoring system for improving student writing through the use of latent semantic analysis*. Doctor of philosophy (thesis), University of Colorado at Boulder.

Wei Wang, Yue Ning, Huzefa Rangwala, and Naren Ramakrishnan. 2016. A multiple instance learning framework for identifying key sentences and detecting events. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 509–518. ACM.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Bronwyn Woods, David Adamson, Shayne Miel, and Elijah Mayfield. 2017. Formative essay feedback using predictive scoring models. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 2071–2080, New York, NY, USA. Association for Computing Machinery.

Qian Yang, Rebecca J Passonneau, and Gerard De Melo. 2016. Peak: Pyramid evaluation via automated knowledge extraction. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Amelia Zafra, Cristóbal Romero, and Sebastián Ventura. 2011. Multiple instance learning for classifying students in learning management systems. *Expert Systems with Applications*, 38(12):15020–15031.