Evaluation of Active Learning for Short Answer Grading

Mohandass Muthuraja, Jeeveswaran Kishaan
M. Sc. Deebul Nair, Prof. Dr. Paul G. Plöger

Hochschule Bonn-Rhein-Sieg

April 10, 2019

Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

Table of Contents

Table of Contents

Introduction

- Assessment of knowledge acquired by the students is one of the important aspect of the learning process.
- Short answer for assessing the knowledge.
  - Self explanation
  - Reasoning
  - Student answers in natural language help in assessing the level of grasping of subject knowledge
- Automatic short answer grading system essentially deals with using computational methods to calculate the grades for students' answers.
- This work is about an AI assisted grading system with a human in the loop.

Motivation and Challenges

Motivation

- Efficient assessment of students' response and providing feedback.
- Digitilization of exams.
- Online learning platforms.
- Grading is subjective in nature which could be assisted.

Challenges

- Short answer grading is not a "learn once and apply forever" task.
- There is no single correct answer for a question. Lexical variations in students' answers need to be captured.
- Cost and time involved in annotating a dataset.

Workflow of Automated Short Answer Grading

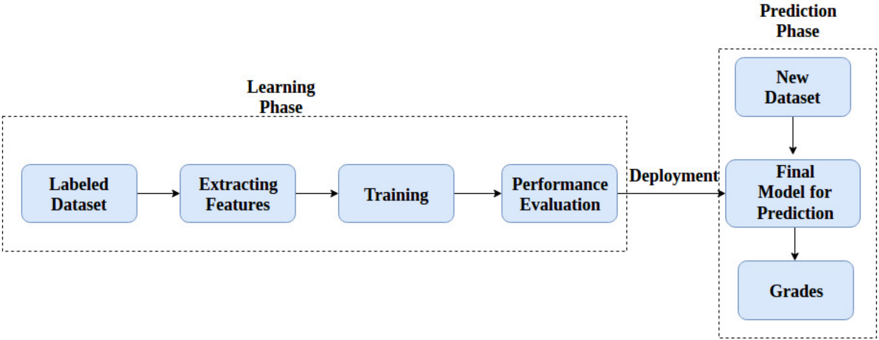


Figure 1: Workflow of automated short answer grading [1]

Workflow of Active Learning in Automated Short Answer Grading

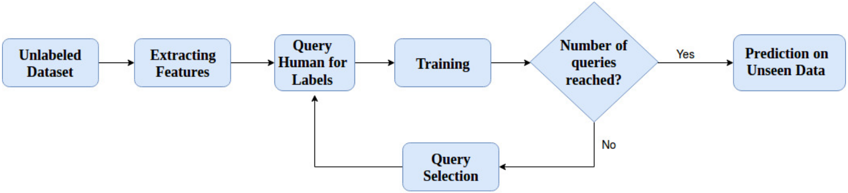


Figure 2: Workflow of active learning in automated short answer grading

Overview

- Active learning have been proved to achieve comparable results with supervised learning with less amount of labeled data in many applications [2] [3].
- The adaptive mechanism of active learning enables the model to learn the new input samples continuously.
- Different active learning settings, features, and machine learning models were evaluated on three different datasets.
- A web-based GUI is designed and implemented to incorporate an AI assisted short answer grading system using the best active learning setting.

Table of Contents

What is Active Learning?

Active learning belongs to a special case of semi-supervised learning algorithm where the learner is allowed to query the user to get the labels for data points which will help the learner to perform better. [4]

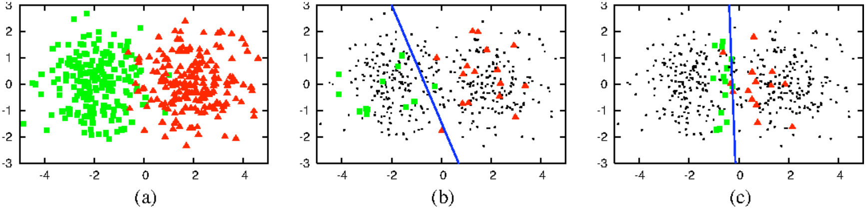


Figure 3: Random sampling vs Active learning. Image from [4]

Table of Contents

Experimental Pipeline

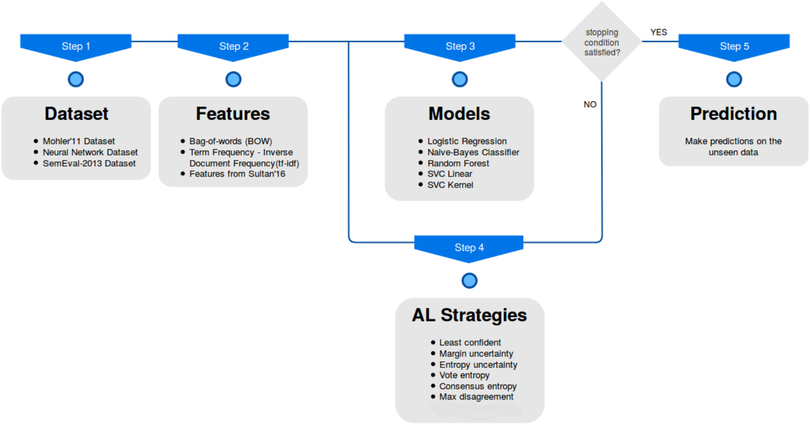
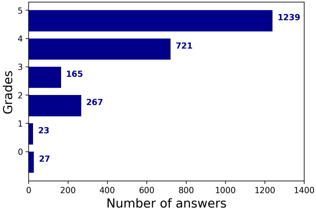


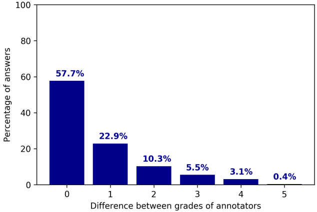
Figure 4: Experiment Pipeline

Mohler'11 Dataset [5]

- 2273 answers from 10 assignments and 2 exams in Computer Science.
- The grades were normalized to 0 to 5 scale.



(a) Grade distribution



(b) Inter-annotator grade analysis [5]

Figure 5: Committee-based binary classification

Neural Network Dataset

- Consists of 646 answers for 17 questions written by 38 students.
- Grades were on a scale of 0 to 2.

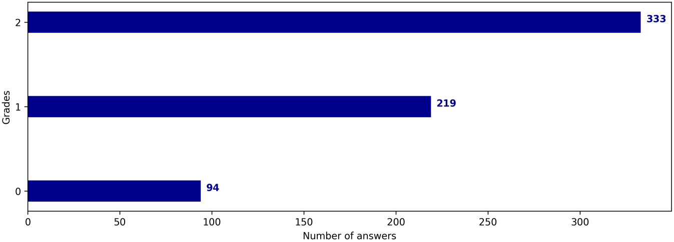


Figure 6: Grade distribution of NN dataset

Datasets

SemEval-2013 Task 7 Dataset [6]

- Grades were on a scale of 0 to 4.
- Dataset was available in four distinct groups namely,
  - Train dataset consist of 4969 answers.
  - 540 unseen answers for the same question.
  - 4562 answers to unseen questions.
  - 733 answers from completely different domain.

Feature Extraction

Pre-processing

- Converting to lowercase
- Removing the punctuations
- Stop words removal
- Lemmatization

Feature Extraction

Bag-of-Words(BOW)

- 'artificial neural network massively parallel distributed processor', 'artificial neural network largely parallel distributed processor', and 'artificial neural network consists neurons'
- ['artificial', 'consists', 'distributed', 'largely', 'massively', 'network', 'neural', 'neurons', 'parallel', 'processor']
- $$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Feature Extraction

Term Frequency-Inverse Document Frequency (Tf-idf)

- Tf-idf is a statistical tool to determine "how important a word is to a document in a collection or corpus" [7].
- Term frequency - This captures the number of occurrence of a word in a document.
- Inverse document frequency - This calculates a low score to frequently occurring words and increasing the weights of the words that occur rarely.

$$tf - idf(t, d) = tf(t, d) \times idf(t) \tag{1}$$

Features from Sultan et al., 2016

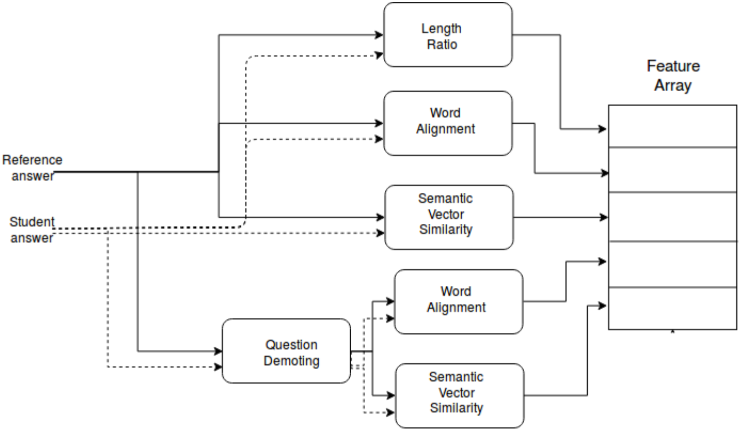


Figure 7: Block diagram of construction of features from Sultam et al.,2016. Image adapted from [8] [9].

Machine Learning Models

- Logistic Regression
- Naive Bayes Classifier
- Random Forests
- Support Vector Machines

Active Learning Query Strategies

Uncertainty Sampling

Instances	Class A	Class B	Class C
I	0.1	0.8	0.1
II	0.35	0.15	0.50
III	0.3	0.3	0.4

Table 1: Prediction probabilty of three instances with respect to three classes

- Least confident uncertainty
  - 0.2, 0.5 and 0.6
- Margin-based uncertainty
  - 0.7,0.15 and 0.1
- Entropy uncertainty
  - 0.64,0.99 and 1.08

Active Learning Query Strategies

Query-by-committee

- Vote entropy

Instances	Model 1	Model 2	Model 3
I	1	1	1
II	2	2	1
III	3	1	1
IV	1	2	3
V	2	2	1

(a) Predicted labels

Instances	Class 1	Class 2	Class 3
I	1	0	0
II	0.3333	0.6667	0
III	0.6667	0	0.3333
IV	0.3333	0.3333	0.3333
V	0.3333	0.6667	0

(b) Class probability distribution

Instances	Entropy
I	0
II	0.6365
III	0.6365
IV	1.0986
V	0.6365

(c) Entropy values

Table 2: Vote entropy

Active Learning Query Strategies

Query-by-committee

- Consensus entropy - Class probability averaged across each learner

Model	Class 1	Class 2	Class 3
Model 1	0.6	0.2	0.2
Model 2	0.5	0.3	0.2
Model 3	0.55	0.35	0.1
Model 4	0.1	0.5	0.4

(a) Class probabilities by every model in first instance.

Model	Class 1	Class 2	Class 3
Model 1	0.3	0.2	0.5
Model 2	0.3	0.5	0.2
Model 3	0.35	0.15	0.5
Model 4	0.2	0.3	0.5

(b) Class probabilities by every model in second instance.

Class 1	Class 2	Class 3	Entropy
0.44	0.34	0.22	1.06

(c) Consensus probability for each class of the first instance.

Class 1	Class 2	Class 3	Entropy
0.29	0.29	0.42	1.08

(d) Consensus probability for each class of the second instance.

Table 3: Consensus entropy

Table of Contents

Results on NN Dataset

Sultan'16 Features with uncertainty sampling query strategy

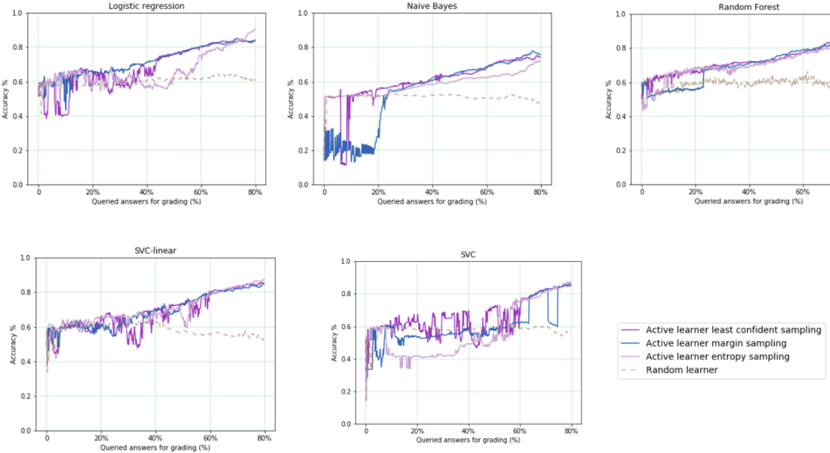


Figure 8: Results for different models with uncertainty based query strategies

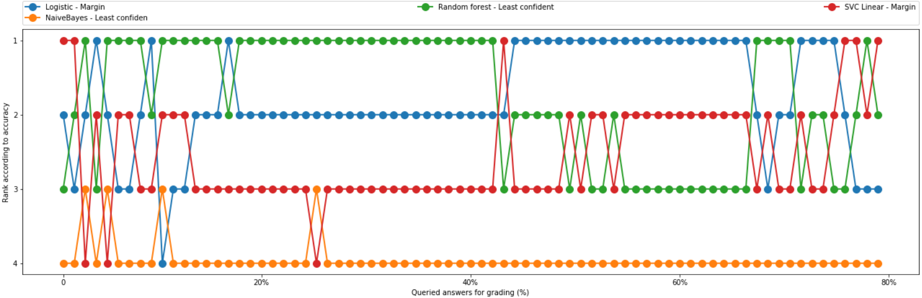


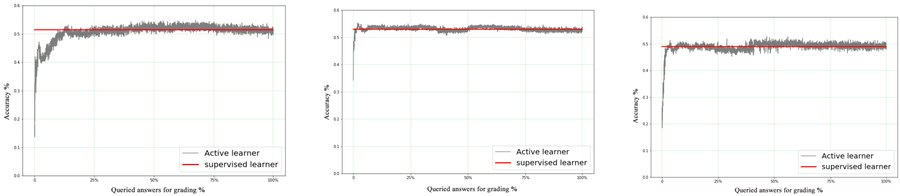
Figure 9: Bump chart of model performance in NN dataset

		Sultan	BOW	TF-IDF
Mohler	Binary	-	Naive Bayes - Margin	Naive Bayes - Margin
	Multi	Random Forest - Least Confident	Naive Bayes - Margin	Naive Bayes - Margin
NN	Multi	Random Forest - Least Confident	Random Forest - Margin	Random Forest - Margin
Sem-Eval	Binary	Logistic Regression - Margin	Random Forest - Margin	Random Forest - Margin
	Multi	Random Forest - Least Confident	Logistic Regression - Margin	Logistic Regression - Margin

Table 4: Best active learning settings on different datasets.

Supervised learning vs Active Learning

- Query strategy: Least confident uncertainty sampling
- Feature: Sultan'16 features
- Model: Random forest classifier



(a) Unseen answers (b) Unseen questions (c) Unseen domain

Figure 10: Comparison of Active learning vs Supervised learning on different datasets

Discussions

- Active learning can reach the same level of performance as supervised learning with less much training data.
- Active learning query strategies outperformed random sampling
- Least confident uncertainty query strategy with Sultan'16 features in random forest classifier performs better than other settings.
  - Features require reference answer for every question.
  - Extracting the features is time-consuming.
- Margin based uncertainty sampling worked well when bag of words or Tf-Idf features were used.
- Batch size of 1 and equal seeding is is found to be in efficient in this task.

Table of Contents

System Architecture

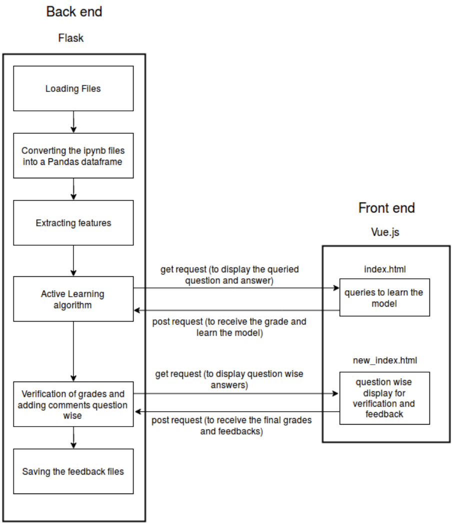


Figure 11: Architecture of the AI-assisted grading system (GUI).

Number of Clicks

- Query Percentage: 25%
- Grading process assisted by active learning massively reduces the effort and time of the grader

Datasets	Clicks with active learning	Clicks without active learning
Neural network	338	680
SemEval 2013	3527	5104
Mohler'11	1386	2352

Table 5: Number of clicks required to grade the answers with and without active learning.

Table of Contents

Contribution

- Detailed evaluation of different active learning settings, features and machine learning model on different datasets.
- Web-based GUI which can be used for the task for grading the answers with the help of active learning.
- Cleaned datasets along with features are available in csv, pandas dataframe formats which could be used in future research works.

Future Work

- A study of features such as sentence embeddings and Latex embedding to improve the performance.
- Efficient active learning strategies to deal with skewed grade distribution in the datasets.
- More functionalities could be added to the GUI and can be integrated with nbgrader.

Acknowledgements

- Active learning framework URL: https://modal-python.readthedocs.io/en/latest/index.html
- Word aligner URL: https://github.com/rameshjesswani/Semantic-Textual-Similarity/tree/master/monolingualWordAligner
- HBRS latex beamer template. URL: https://git.fslab.de/mmklab/latex-templates/tree/master/presentation

References

[1] S. Burrows, I. Gurevych, and B. Stein. The eras and trends of automatic short answer grading. Vol. 25. 1. 2015, pp. 60–117. ISBN: 4059301400268. DOI: 10.1007/s40593-014-0026-8.

[2] D. Dligach and M. Palmer. “Good seed makes a good crop: accelerating active learning using language modeling”. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2. Association for Computational Linguistics. 2011, pp. 6–10.

[3] R. L. Figueroa et al. “Active learning for clinical text classification: is it better than random sampling?” In: Journal of the American Medical Informatics Association 19.5 (2012), pp. 809–816.

[4] B. Settles. “Active Learning Literature Survey”. In: Machine Learning 15.2 (2010), pp. 201–221. ISSN: 00483931. DOI: 10.1.1.167.4245. arXiv: 1206.5533.

[5] M. Mohler, R. Bunescu, and R. Mihalcea. “Learning to grade short answer questions using semantic similarity measures and dependency graph alignments”. In: (2011), pp. 752–762.

[6] M. O. Dzikovska et al. Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. Tech. rep. NORTH TEXAS STATE UNIV DENTON, 2013.

[7] tf-idf. https://en.wikipedia.org/wiki/Tf-idf. Accessed: 2018-11-22.

[8] “Evaluation of Semantic Textual Similarity Approaches for Automatic Short Answer Grading”. WS17 H-BRS - Evaluation of Semantic Textual Similarity Approaches for Automatic Short Answer Grading Ploeger, Nair supervising. 2017/18.

[9] M. A. Sultan, C. Salazar, and T. Sumner. “Fast and Easy Short Answer Grading with High Accuracy”. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2016), pp. 1070–1075. DOI: 10.18653/v1/N16-1123.

Demo


Hochschule
Bonn-Rhein-Sieg
Autograder$_{\text{beta}}$
Question :
Define the mathematical model of a neuron, use the appropriate technical terms!
Answer :
N number inputs, x_i input i, v_j local field, $\varphi(v_j)$ activiation function, y_j output, w_{ji} weight from node i to j $y_j = \varphi(v_j)$ $v_j = \sum_{i=0}^{N} w_{ji} x_i$
Manually graded Score : 0 1 2
Submit

Thank you!

Questions?