ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-1622

Transfer Learning for Automatic Short Answer Grading

Shourya Roy[1] and **Himanshu S. Bhatt**[1] and **Y. Narahari**[2]

Abstract. Automatic short answer grading (ASAG) is the task of automatically grading students answers which are a few words to a few sentences long. While supervised machine learning techniques (classification, regression) have been successfully applied for ASAG, they suffer from the constant need of instructor graded answers as labelled data. In this paper, we propose a transfer learning based technique for ASAG built on an ensemble of text classifier of student answers and a classifier using numeric features derived from various similarity measures with respect to instructor provided model answers. We present preliminary empirical results to demonstrate efficacy of the proposed technique.

1 Introduction

Assessment of student answers constructed in natural language has remained predominantly a manual job owing to multiple reasons. These include *linguistic variations* (same answer could be articulated in different ways); *subjective nature of assessment* (multiple possible correct answers or no correct answer); *lack of consistency in human rating*; etc. This paper dwells on a computational technique for automatically grading constructed student answers in natural language by focusing on *short answers*: a few words to a few sentences long [6]) and refer to the task as *Automatic Short Answer Grading (ASAG)*. Data for an example ASAG task is shown in Table 1 where the task is to automatically score the student answers.

In the next section, we intuitively describe the approach taken in this work towards developing a transfer learning based ASAG technique. In Section 3, we will describe the solution in greater detail followed by preliminary empirical evidence of benefits of the proposed approach.

Question	How are overloaded functions differentiated by the compiler? (5)
Model Ans	Based on the function signature. When an overloaded function is called, the compiler will find the function whose signature is closest to the given function call.
Stud#1	it looks at the number, types, and order of arguments in the function call.
Stud#2	By the number, and the types and order of the parameters.

Table 1. Example of question, model answer, and student answers from an undergraduate computer science course [3].

2 Approach

A large fraction of prior work in ASAG has been based on supervised learning techniques viz. classification and regression [4]. These techniques utilize features extracted from student and model answers using natural language processing (NLP) techniques reflecting *similarity* (synonymously, *overlap*, *correspondence*, *entailment* etc.) between them. These features are then fed to various classification or regression techniques to train models which can subsequently be applied to score new student answers automatically. In this work, we propose a novel supervised ASAG technique based on an ensemble of two classifiers. The first of the two is a text classifier trained using the classical TFIDF representation of bag-of-words features of student answers. It is independent of model answers and learns good textual features (words and n-grams) from graded student answers to discriminate between student answers belonging to different scores. On the other hand, features of the second classifier are a bunch of real numbers indicating similarity of student answers with the corresponding model answer. While both classifiers are trained for the same task of predicting scores for new student answers, they leverage different sets of features. Additionally, they are complimentary in nature owing to their independence and dependence on model answers respectively.

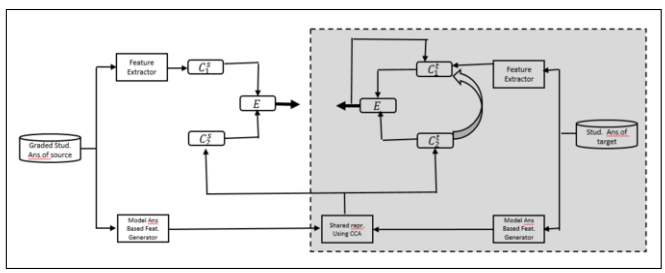

Figure 1. Block diagram of the proposed algorithm. The shaded part can be replicated for target questions for which no labelled data is available.

Supervised ASAG techniques require ongoing instructor involvement to create labelled data (by grading $\frac{1}{2}$ to $\frac{2}{3}$ of student answers as per typical train-test split) for every question and assessment task. Requirement of such continuous involvement of instructor limits the benefit of automation and thereby poses a hindrance to practical adoption. Towards addressing this limitation, we bring in the concept of *transfer learning*. Transfer learning techniques work, in contrast to traditional supervised techniques, on the principle of transferring learned knowledge across *domains*. These techniques learn a supervised model on a *source* domain with plenty of labelled data and apply to another *target* domain without (or minimal) labelled data. We formulate ASAG as a transfer learning task by considering answers to different questions as different domains as they have different marginal probability distributions of features and possibly different features too. Towards that, we propose a feature transformation based transfer learning technique using canonical correlation analysis (CCA) between the source and target questions. It transfers the trained model of the second classifier of the source question ensemble by learning a common shared representation of features which minimizes domain divergence and classification error.

[1] Xerox Research Centre India, Bangalore, India email: firstname.lastname@xerox.com
[2] Indian Institute of Science, Bangalore, India email: hari@csa.iisc.ernet.in

3 Technique

Figure 1 shows the key components of the proposed system. In this section we describe the same highlighting two main themes:

Ensemble of classifiers: We model ASAG as a supervised learning task where we employ an ensemble of two classifiers to predict student scores. The first classifier (C_1) uses the popular TFIDF vectorization on bag-of-word representations of student answers and convert to TFIDF vectors with corresponding grades as class labels. Prior to vectorization, we perform basic NLP pre-processing of stemming and stopword removal. We also perform question word demoting (i.e. considering words appearing in the question as stopwords while vectoring student answers) to avoid giving importance to parrot answering. The second classifier (C_2) is based on real-valued features capturing similarity of student answers with respect to model answer. In our endeavor towards generalizability of the proposed technique, we employ multiple generic state of the art measures to compute similarity between two pieces of short text (model and student answers) covering *lexical* (BLEU [5]), *semantic* (Wordnet based measures [3]) and *vector-space* measures (latent semantic analysis and word vectors [2]). Additionally, we would like readers to note that model of the first classifier is question specific (i.e. a word which is a good feature for a question is not necessarily a good feature for another question), whereas features for the second classifier are more question agnostic (i.e. high similarity with *respective* model answer is indicative of high scores irrespective of question). Finally, these two classifiers are combined in a weighted manner to form an ensemble (E) which is used for enhanced automatic short answer grading.

Transfer based on common representation: The ensemble of classifiers can be developed as described above for the source question based on instructor graded answers. The question is how do we do the same for target questions in absence of graded answers? It is done in two steps - (i) obtaining the second classifier through a common feature space based transfer of model from source to target followed by (ii) iteratively building the first classifier and the ensemble using pseudo labeled data.

Learning a common representation for ASAG task is based on finding a shared projection of the question agnostic features (used in the second classifier) from source and target questions. For numeric features, we used the classical canonical correlation analysis (CCA) [1] which extracts features from source and target questions such that the projected features from the two becomes maximally correlated. It learns multiple projection vectors to transform the real valued features from the source and target questions respectively to have maximum correlation. The source labeled instances are then projected onto a subspace (with the learnt projection vectors as bases) to learn a model which is subsequently used to predict labels of target instances in this subspace.

The newly trained classifier on CCA-based transformed features is the second classifier of target question. It is applied to all student answers to target question and *confidently* predicted answers are chosen as pseudo-labeled data to train the first version of the first classifier of the target question. We call this training data pool as pseudo as these are not labeled by the instructor rather based on (confident) predictions of the second classifier. This, along with the transferred second classifier are combined as an ensemble (as described above) and tested on the remaining student answers (i.e. which were not in pseudo labeled training data). Confidently predicted instances from the ensemble are subsequently iteratively used to re-train the text classifier and boost up the overall prediction accuracy of the ensemble. The iteration continues till all the examples are correctly predicted or a specified number of iterations are performed.

4 Evaluation

We empirically evaluated the proposed technique on a dataset from an undergraduate computer science course (CSD) [3] and one of its extended version (X-CSD). They consist of 21 and 87 questions respectively from introductory assignments in the course with answers provided by a class of abut 30 undergraduate students. We followed the convention in transfer learning literature of comparing against a skyline and a baseline:

- **Baseline (Sup-BL)**: Supervised models are built using labeled data from a source question and applied *as-it-is* to a target question.
- **Skyline (Sup-SL)**: Supervised models are built assuming labeled data is available for all questions (including target). Performance is measured by training a model on every question and applied on the same.

Performances of transfer learning techniques should be in between the baseline and skyline - closer to the skyline, better it is.

We use mean absolute error (MAE) as the metric for quantitative evaluation. MAE for a question is the absolute difference between groundtruth and predicted scores averaged over all students ($\frac{1}{n}\sum_{i=1}^{n}|t_i - y_i|$), where t_i and y_i are respectively the groundtruth and predicted scores of the i^{th} student's answer. For reporting one number for the dataset, the values are averaged for all questions.

Aggregated performances of ASAG techniques for the three datasets are shown in Table 3.

	CSD	X-CSD
Sup-BL	2.46	4.52
Sup-SL	0.64	0.92
Proposed	0.81	1.41

Table 2. Overall performance (MAE) of the proposed algorithm grading along with the baseline and skyline (lower the better).

The proposed method beats the baseline for both datasets handsomely (differences being 1.65 and 3.11) whereas coming much closer to the skyline (differences being 0.17 and 0.49). This demonstrates benefit of the proposed technique over supervised learning based ASAG techniques.

REFERENCES

[1] C.H. Huang, Y. R. Yeh, and Y. C. F. Wang, 'Recognizing actions across cameras by exploring the correlated subspace', in *Proceedings of International Conference on Computer Vision - Volume Part I*, pp. 342–351, (2012).

[2] T. Mikolov, K. Chen, G. Corrado, and J. Dean, 'Efficient estimation of word representations in vector space', *arXiv preprint arXiv:1301.3781*, (2013).

[3] M. Mohler and R. Mihalcea, '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*, pp. 567–575, (2009).

[4] Michael Mohler, Razvan C. Bunescu, and Rada Mihalcea, 'Learning to grade short answer questions using semantic similarity measures and dependency graph alignments.', in *ACL*, pp. 752–762, (2011).

[5] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu, 'Bleu: a method for automatic evaluation of machine translation', Technical report, IBM Research Report, (2001).

[6] Shourya Roy, Y Narahari, and Om D Deshmukh, 'A perspective on computer assisted assessment techniques for short free-text answers', in *Computer Assisted Assessment. Research into E-Assessment*, 96–109, Springer, (2015).