Context Based Classification for Automatic Collaborative Learning Process Analysis

Yi-Chia WANG[1], Mahesh JOSHI[1], Carolyn ROSÉ[1], Frank FISCHER[2], Armin WEINBERGER[2], Karsten STEGMANN[2]
[1]*Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA, 15213 USA*
{yichiaw,maheshj,cp3a}@andrew.cmu.edu
[2]*Ludwig Maximillians Universitaet, Leopoldstrasse 13, 80802 Munich, Germany*
{Frank.Fischer,Armin.Weinberger,Karsten.Stegmann}@psy.lmu.de

Abstract. We present a publicly available tool called TagHelper that can be used to support the analysis of conversational data using automatic text classification technology. The contribution of this paper is to explore the limitations of the current simple approach to text processing employed by TagHelper tools with respect to identifying context-sensitive categories of conversational behavior. TagHelper can be downloaded from http://www.cs.cmu.edu/~cprose/TagHelper.html.

Keywords. Collabortive Learning, Automatic Process Analysis

Introduction

Building on earlier work [1], we present a publicly available tool called TagHelper tools that can be used to automate the analysis of conversational data using machine learning technology. Its goal is to facilitate the process of investigating which conversational processes are effective for supporting learning. Conversational process analysis is an especially important area in the field of Computer Supported Collaborative Learning (a review of this area of research is provided in an earlier publications [1,2]). Ultimately, beyond facilitating the study of learning processes in human tutoring and collaborative learning, automatic process analyses can also be used to trigger interventions during learning interactions.

We begin by describing the functionality that TagHelper tools makes available to researchers. We then discuss some limitations of the current version of TagHelper tools and how we are addressing them in our ongoing research. We conclude with some lessons learned, which can by applied by other researchers exploring the use of machine learning technology for supporting analysis of conversational processes in a learning context.

The research contribution of this paper is to explore the limitations of the current approach to text processing employed by TagHelper tools specifically with respect to identifying context-sensitive categories of conversational behavior, which are important for characterizing ongoing interactions between two or more people. In other words, a span of text may be ambiguous with respect to its conversational function when taken out of context. If each span of text is processed in isolation, the information required to disambiguate the function of such spans of text is not available. We present a series of investigations using a corpus of newsgroup style conversational interactions collected during a series of computer supported collaborative learning studies, which has been annotated with the coding scheme described in [2]. We describe our work in progress towards using context to more robustly identify context sensitive categories of conversational behavior.

1. TagHelper Tools

TagHelper's basic classification functionality is now available to researchers to use in their own work. Because we have observed the popular usage of Microsoft Excel as an environment in which behavioral researchers commonly do their corpus analysis work, we have adopted that as our standard file format. The TagHelper application and documentation can now be downloaded free of charge from http://www.cs.cmu.edu/~cprose/TagHelper.html. Note that this version of TagHelper that is publicly available is designed to work with presegmented English, German, or Chinese texts. In order to make TagHelper tools accessible to the widest possible user base, we have set up its default behavior in such a way that users are only required to provide examples of annotated data along with un-annotated data. At the click of a button, TagHelper tools builds a model based on the annotated examples that it can then apply to the un-annotated examples. More advanced users can experiment with a variety of available settings, which may allow them to achieve better performance with TagHelper tools. An analyst has two types of options in customizing the behavior of TagHelper. One is to manipulate the structured representation of the text, and the other is to manipulate the selection of the machine learning algorithm. These two choices are not entirely independent of one another. An insightful machine learning practitioner will think about how the representation of their data will interact with the properties of the algorithm they select. Interested readers are encouraged to read Witten & Frank's (2005) book, which provides a comprehensive introduction to the practical side of the field of machine learning. In the near future we hope to provide an on-line training course to interested practitioners.

2. Context Based Approach

Here we discuss our investigations for extending the capabilities of TagHelper tools with respect to coding schemes that rely on judgments about how spans of text relate to the context in which they appear. In our coding methodology, we first segment a message into spans of text referred to as epistemic units [2]. Each of these units of text is then assigned one code from each of seven dimensions in our multi-dimensional coding scheme. In this paper we focus on a single one of these dimensions referred to as the Social Modes of Co-Construction, which is highly context sensitive. This dimension indicates to what degree or in what ways learners refer to the contributions of their learning partners. In this dimension there are five types of social modes, namely externalizations, elicitation, quick consensus building, integration building, and conflict-oriented consensus building. There is also a category for "other" contributions. Thus, with respect to the Social Modes of Co-Construction, each message potentially has a sequence of several codes, one for each unit of text. Furthermore, using the natural structure of a threaded discussion, we can automatically extract some features that provide some clues about the discourse function of a span of text. Thus, there are two sources of context information in the structured data that we have that we make use of. First, there is the course grained thread structure, with parent-child relationships between messages. And secondly, there is the sequence of codes that are assigned to units of text within a message.

We refer to features that are related to the threaded structure of the message board as *thread structure features*. The simplest such feature is a number indicating the depth in the thread where a message appears, which is called *deep*. This is expected to improve performance somewhat since some codes within the coding scheme never appear in thread initial messages. Other context oriented features related to the thread structure are derived from relationships between spans of text appearing in the parent and child messages. One such feature indicates how semantically related a span of text is to the spans of text in the parent message. This is computed using the cosine distance between the vector representation of the span of text and that of each of the spans of text in the parent message. The smallest such distance is included as a feature.

Next we have *sequence oriented features*. We hypothesized that the sequence of codes within a message follows a semi-regular structure. In particular, the CSCL environment inserts prompts into the message buffer that students use. Students fill in text underneath these prompts. Sometimes they quote material from a previous message before inserting their own comments. We hypothesized that whether or not a piece of quoted material appears before a span of text might influence which code is appropriate. Thus, we constructed the *fsm* feature, which indicates the state of a simple automaton, which only has two states. The automaton is set to initial state (q0) at the top of a message. It makes a transition to state (q1) when it encounters a quoted span of text. Once in state (q1), the automaton remains in this state until it encounters a prompt. On encountering a prompt it makes a transition back to the initial state (q0).

The purpose of our evaluation is to contrast our proposed feature based approach with a state-of-the-art sequential learning technique [3]. Both approaches are designed to leverage context for the purpose of increasing classification accuracy on a classification task where the codes refer to the role a span of text plays in context. We first evaluate alternative combinations of features using SMO, Weka's implementation of Support Vector Machines [4]. For a sequential learning algorithm, we make use of the Collins Perceptron Learner [3]. When using the Collins Perceptron Learner, in all cases we evaluate combinations of alternative history sizes (0 and 1) and alternative feature sets (base and base+AllContext). In our experimentation we have evaluated larger history sizes as well, but the performance was consistently worse as the history size grew larger then 1. Thus, we only report results for history sizes of 0 and 1. Our evaluation demonstrates that we achieve a greater impact on performance with carefully designed, automatically extractable context oriented features.

We first evaluated the feature based approach. The results demonstrate that statistically significant improvements are achieved by adding context oriented features (See Figure 1). All pairwise contrasts between alternative feature sets are statistically significant. The results for sequential learning are much weaker than for the feature based approach. First, the Collins Perceptron learner consistently performs significantly worse that SMO with or without context features. However, even restricting our evaluation of sequential learning to a comparison between the Collins Perceptron learner with a history of 0 (i.e., no history) with the same learner using a history of 1, we only see a statistically significant improvement on the Social Modes of Co-Construction dimension using only base features. Note that the standard deviation in the performance across folds was much higher with the Collins Perceptron learner, so that a much greater difference in average would be required in order to achieve statistical significance. Nevertheless, the trend was consistently in favor of a history of 1 over a history of 0. Thus, there is some evidence that sequential learning provides some benefit. Performance was always worse with larger history sizes than 1.

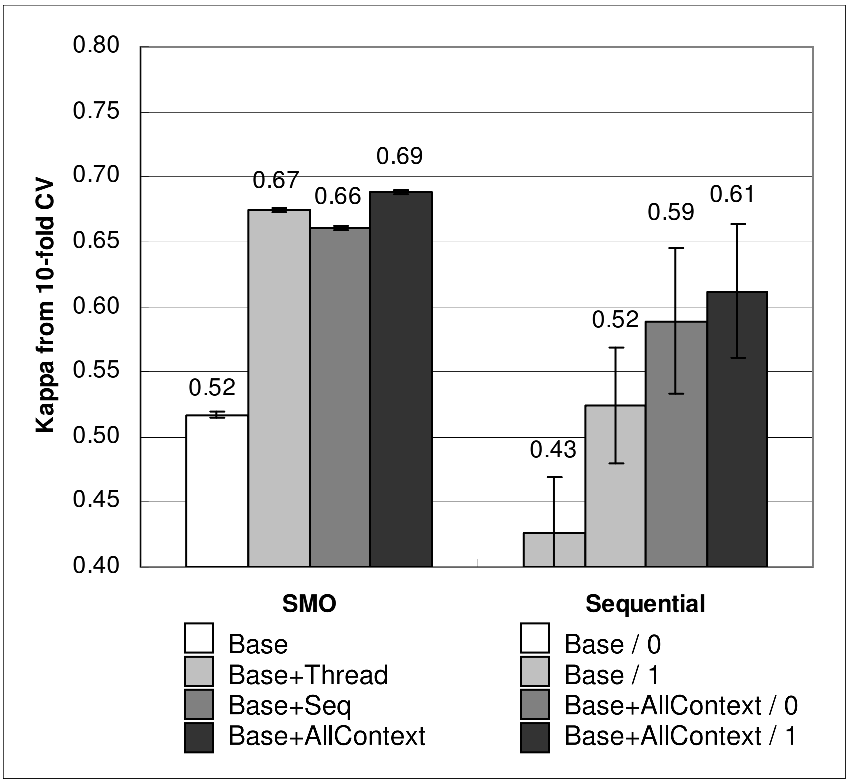

Figure 1 Performance of two classification algorithms (SMO vs Collins Perceptron) using 4 different feature sets.

3. Conclusions

We have proposed and evaluated two differed approaches for improving classification performance with a context oriented coding scheme. We showed that our selected context oriented features indeed can improve the performance of various learning algorithms, including both non-sequential and sequential ones. However, we did not observe an increase in performance due to using a sophisticated sequential learning algorithm such as the Collins Perceptron Learner. We believe the important take home message is that effectively applying machine learning to the problem of automatic collaborative learning process analysis requires selecting appropriate features that can approximate the linguistic mechanisms that underly the design of the categorical coding scheme that is used.

This project is supported by ONR Cognitive and Neural Sciences Division, Grant number N000140510043 and NSF Grant number SBE0354420.

References

[1] Donmez, P., Rose, C. P., Stegmann, K., Weinberger, A., and Fischer, F. Supporting CSCL with Automatic Corpus Analysis Technology, *Proceedings of Computer Supported Collaborative Learning* (2005).

[2] Weinberger, A., & Fischer, F. A framework to analyze argumentative knowledge construction in computer-supported collaborative learning. *Computers & Education*, 46, 71-95 (2006).

[3] Witten, I. H. & Frank, E. *Data Mining: Practical Machine Learning Tools and Techniques*, second edition, Elsevier: San Francisco, ISBN 0-12-088407-0 (2005).

[4] Michael Collins. Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. In *Proceedings of Empirical Methods for Natural Language Processing 2002*. (2002)