Argument Component Classification for Classroom Discussions

Luca Lugini and **Diane Litman**
Computer Science Department & Learning Research and Development Center
University of Pittsburgh
Pittsburgh, PA 15260
{lul32,dlitman}@pitt.edu

Abstract

This paper focuses on argument component classification for transcribed spoken classroom discussions, with the goal of automatically classifying student utterances into claims, evidence, and warrants. We show that an existing method for argument component classification developed for another educationally-oriented domain performs poorly on our dataset. We then show that feature sets from prior work on argument mining for student essays and online dialogues can be used to improve performance considerably. We also provide a comparison between convolutional neural networks and recurrent neural networks when trained under different conditions to classify argument components in classroom discussions. While neural network models are not always able to outperform a logistic regression model, we were able to gain some useful insights: convolutional networks are more robust than recurrent networks both at the character and at the word level, and specificity information can help boost performance in multi-task training.

1 Introduction

Although there is no universally agreed upon definition, argument mining is an area of natural language processing which aims to extract structured knowledge from free-form unstructured language. In particular, argument mining systems are built with goals such as: detecting what parts of a text express an argument component, known as argument component identification; categorizing arguments into different component types (e.g. claim, evidence), known as argument component classification; understanding if/how different components are connected to form an argumentative structure (e.g. using evidence to support/attack a claim), known as argument relation identification. The development and release to the public of corpora and annotations in recent years have contributed to the increasing interest in the area.

One domain in which argument mining is rarely found in the literature is educational discussions. Classroom discussions are a part of students' daily life, and they are a common pedagogical approach for enhancing student skills. For example, student-centered classroom discussions are an important contributor to the development of students' reading, writing, and reasoning skills in the context of English Language Arts (ELA) classes (Applebee et al., 2003; Reznitskaya and Gregory, 2013). This impact is reflected in students' problem solving and disciplinary skills (Engle and Conant, 2002; Murphy et al., 2009; Elizabeth et al., 2012). With the increasing importance of argumentation in classrooms, especially in the context of student-centered discussions, automatically performing argument component classification is a first step for building tools aimed at helping teachers analyze and better understand student arguments, with the goal of improving students' learning outcomes.

Many current argument mining systems focus on analyzing argumentation in student essays (Stab and Gurevych, 2014, 2017; Nguyen and Litman, 2015, 2018), online dialogues (Swanson et al., 2015; McLaren et al., 2010; Ghosh et al., 2014; Lawrence and Reed, 2017), or in the legal domain (Ashley and Walker, 2013; Palau and Moens, 2009). A key difference between these studies and our work consists in the source of linguistic content: although we analyze written transcriptions of discussions, the original source for our corpora consists of spoken, multi-party, educational discussions, and the difference in cognitive skills and grammatical structure between written and spoken language (Biber, 1988; Chafe and Tannen, 1987) introduces additional complexity.

Our work and previous research studies on student essays share the trait of analyzing argumentation in an educational context. However, while

student essays are typically written by an individual student, in classroom discussions arguments are formed collaboratively between multiple parties (i.e. multiple students and possibly teachers). While our work shares the multi-party context in which arguments are made with research aimed at argument mining in online dialogues, prior online dialogue studies have not been contextualized in the educational domain.

Given these differences, we believe that argument mining models for student essays and online dialogues will perform poorly when directly applied to educational discussions. However, since similarities between the domains do exist, we expect that features exploited by such argument mining models can help us in classifying argument components in classroom discussions. Moreover, unlike the other two domains, we have access to labels belonging to a different (but related) class, specificity, which we can try to incorporate in argumentation models to boost performance.

Our contributions are as follows. We first experimentally evaluate the performance of an existing argument mining system developed for essay scoring (named wLDA) when applied off-the-shelf to predict argument component labels for transcribed classroom discussions. We then analyze the performance obtained when using the same features as wLDA to train a classifier specifically on our dataset. We combine the wLDA feature set with features used in argument mining in the context of online dialogues and show that they are able to capture some of the similarities between online dialogues and our domain, and considerably improve the model. We then evaluate two neural network models in several different scenarios pertaining to their input modality, the inclusion of handcrafted features, and the effect of multi-task learning when including specificity information.

2 Related Work

With respect to the educational domain, previous studies in argument mining were largely aimed at student essays. Persing and Ng (2015) studied argument strength with the ultimate goal of automated essay scoring. Stab and Gurevych (2014) performed argument mining on student essays by first jointly performing argument component identification and classification, then predicting argument component relations. Nguyen and Litman (2015) developed an argument mining system for analyzing student persuasive essays based on argument words and domain words. While domain words are used only in a specific topic, argument words are used across multiple topics and represent indicators of argumentative content. They later proposed an improved version of the system (2016), which we will refer to as wLDA, by exploiting features able to abstract over specific essay topics and improve cross-topic performance. While our current work is also aimed at developing argument mining systems in the educational context, we focus on educational discussion instead of student essays. Our work also differs in the argument component types used: we analyze claims, evidence, and warrants, while prior studies mostly focused on claims and premises. The inclusion of warrants is particularly important to explicitly understand how students use them to connect evidence to claims. As such, we do not expect prior models to work well on our corpus, although some of the features might still be useful. Also, while some of the previously proposed systems address multiple subproblems simultaneously, e.g. argument component identification and argument component classification, we only focus on argument component classification.

Swanson et al. (2015) developed a model for extracting argumentative portions of text from online dialogues, which were later used for summarizing the multiple argument facets. Misra et al. (2015) analyzed dyadic online forum discussions to detect central propositions and argument facets. Habernal and Gurevych (2017) analyzed user-generated web discourse data from several sources by performing micro-level argumentation mining. While these prior works analyze multi-party discussions, the discussions are neither originally spoken nor in an educational context.

Like other areas of natural language processing, argument mining is experiencing an increase in the development of neural network models. Niculae et al. (2017) used a factor graph model which was parametrized by a recurrent neural network. Daxenberger et al. (Daxenberger et al., 2017) investigated the different conceptualizations of claims in several domains by analyzing in-domain and cross-domain performance of recurrent neural networks and convolutional neural networks, in addition to other models. Schulz et al. (Schulz et al., 2018) analyzed the impact of using multi-task learning when training on a limited amount of labeled data. In a similar way, we develop several convolutional neural network and recurrent neural

network models, and also experiment with multi-task learning. More detailed comparisons will be given in Section 4.

3 Dataset

We collected 73 transcripts of text-based classroom discussions, i.e. discussions centered on a text or literature piece (e.g. play, speech, book), for ELA high school level classes. Some of the transcripts were gathered from published articles and dissertations, while the rest originated from videos which were transcribed by one of our annotators (see below). While detailed demographic information for students participating in each discussion was not available, our dataset consists of a mix of small group (16 out of 73) versus whole class (57/73) discussions, both teacher-mediated (64/73) versus student only (9/73). Additionally, the discussions originated in urban schools (28/73), suburban schools (42/73), and schools located in small towns (3/73).

The unit of analysis for our work is argument move, which consists of a segment of text containing an argumentative discourse unit (ADU) (Peldszus and Stede, 2013). Starting with transcripts broken down into turns at talk, an expert annotator segmented turns at talk into multiple argument moves when necessary: turns at talk containing multiple ADUs have been segmented into several argument moves, each consisting of a single ADU. Turn segmentation effectively corresponds to argument component identification, and it is carried out manually. We conducted a reliability study on turn segmentation with two annotators on a subset of the dataset consisting of 53 transcripts. The reliability analysis resulted in Krippendorff $\alpha_U = 0.952$ (Krippendorff, 2004), which shows that turns at talk can be reliably segmented.

After segmentation, the data was manually annotated to capture two aspects of classroom talk, argument component and specificity, using the ELA classroom-oriented annotation scheme developed by Lugini et al. (2018). The argument component types in this scheme, which is based on the Toulmin model (1958), are: (i) *Claim*: an arguable statement that presents a particular interpretation of a text or topic. (ii) *Evidence*[1]: facts, documentation, text reference, or testimony used to support or justify a claim. (iii) *Warrant*: rea-

sons explaining how a specific evidence instance supports a specific claim.

Chisholm and Godley (2011) observed how specificity has an impact on the quality of the discussion, while Swanson et al. (2015) noted that a relationship exists between specificity and the quality of arguments in online forum dialogues. For the purpose of investigating whether there exists a relationship between specificity and argument components, we additionally annotated data for specificity following the same coding scheme (Lugini et al., 2018). Specificity labels are directly related to four elements for an argument move: (1) it is specific to one (or a few) character or scene; (2) it makes significant qualifications or elaborations; (3) it uses content-specific vocabulary (e.g. quotes from the text); (4) it provides a chain of reasons. The specificity annotation scheme by Lugini et al. includes three labels along a linear scale: (i) *Low*: statement that does not contain any of these elements. (ii) *Medium*: statement that accomplishes one of these elements. (iii) *High*: statement that clearly accomplishes at least two specificity elements. Only student turns were considered for annotations; teacher turns at talk were filtered out and do not appear in the final dataset. Table 1 shows a coded excerpt of a transcript from a discussion about the movie *Princess Bride*.

The resulting dataset consists of 2047 argument moves from 73 discussions. As we can see from the label distribution shown in Table 2, students produced a high number of claims, while warrant is the minority class. We can also observe a class imbalance for specificity labels, though the ratio between majority and minority classes is lower than that for argument component labels.

We evaluated inter-rater reliability on a subset of our dataset composed of 1049 argument moves from 50 discussions double-coded by two annotators. Cohen's unweighted kappa for argument component labels was 0.629, while quadratic-weighted kappa for specificity labels (since they are ordered) was 0.641, which shows substantial agreement.

The average number of argument moves among the discussions is 27.3 while the standard deviation is 25.6, which shows a high variability in discussion length. The average number of words per argument move and standard deviation are 22.6 and 22.1, respectively, which also shows large variability in how much students speak.

---

[1]The "evidence" label is equivalent to "data" or "grounds" used in the original Toulmin model, though we use the label "evidence" to remain consistent with the annotation scheme.

Stu	Argument Move	Arg Comp	Spec
S1	Well Fezzik went back to how he was,	Claim	Low
S1	like how he gets lost. Then he goes like he needs to be around other people. And then finally when he does, he gets himself like relying on himself. But then right at the end, he doesnt know where hes at; he makes a wrong turn.	Evidence	Med
S1	cause he tried doing it by himself and he cant. So I think Fezzik went back to his normal ways, like after he changed.	Warrant	High

Table 1: Coded excerpt of a discussion of the movie *Princess Bride*. Student S1 first makes a claim about Fezzik's behavior, then provides evidence by listing a series of events, then connects such events to his claim using a warrant. As the argument progresses, the specificity level increases.

Argument Component		
Claim	Warrant	Evidence
1034	358	655
Specificity		
Low	Med	High
710	996	341

Table 2: Distribution of class labels for argument component type and specificity in our dataset.

4 Argument Component Classification

In this section we outline an existing argument component classification system that will serve as a baseline for our experiments, then propose several new models that use features extracted from neural networks and hand-crafted features, as well as models that use multi-task learning.

4.1 Existing Argument Mining System

The wLDA[2] system was developed for performing argument component identification, classification, and relation extraction from student essays. For the purpose of this study, we only consider the argument component classification subsystem. The model is based on a support vector machine classifier which exploits features able to improve cross-topic performance. The feature set consists of four main subsets: lexical features (argument words, verbs, adverbs, presence of modal verbs, discourse connectives, singular first person pronoun); parse features (argumentative subject-verb pairs, tense of the main verb, number of sub-clauses, depth of parse tree); structural features (number of tokens, token ratio, number of punctuation signs, sentence position, first/last paragraph, first/last sentence of paragraph); context features (number of tokens, number of punctuation signs, number of

sub-clauses, modal verb in preceding/following sentences) extracted from the sentences before and after the one considered; four additional features for abstracting over essay topics.

Since the model was trained on essays annotated for major claim, claim, and premise, but not on warrants, in our evaluation we did not take into account misclassification errors for argument moves in our dataset labeled as warrants. The pre-trained system performs argument component identification using a multiclass classification approach, such that each input will be classified as non argumentative, major claim, claim or premise. Since our goal is to evaluate performance related to the component classification problem, we ignored all the argument moves classified as non argumentative by wLDA. Considering the definitions of premise and evidence in the Toulmin model (1958), we made the assumption of the two labels being equivalent for this study, i.e. if the predicted class for an argument move is premise and its gold standard label in our dataset is evidence, we consider the prediction correct. In the same way we consider both claim and major claim labels as equivalent to claims in our dataset.

4.2 Neural Network Models

Since the pre-trained model did not work well on our dataset, and the features it is based on show a large gap in performance compared to the original work (see Section 5), we decided to use neural networks, and evaluate their ability to automatically extract meaningful features. The proposed models consist of variations of two basic neural network models, namely Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) models. All the choices regarding the models were made in order to keep complexity and the number of weights at a minimum, since neural net-

---

[2]The original name of wLDA+4 stands for "with LDA supported features and expanded with 4 features sets" compared to their previous system. We use wLDA for brevity.

work models require in general a large amount of training data, while we have a limited size dataset. The CNN model is based on a model proposed by Kim (2014) and already used for argument mining in the past (Daxenberger et al., 2017), with a difference in the number of convolutional/pooling layers. In particular, our model uses 3 convolutional/max pooling layers instead of 6, and only one fully connected layer after the convolutional ones, followed by a softmax layer used for classification. This choice resulted from observing significant overfitting when increasing the number of convolutional layers due to the increase in the number of model weights and the limited dataset size. Figure 1 shows diagrams for the different neural network setups used in our experiments.

The RNN model consists of a single Long-Short Term Memory (LSTM) network (Hochreiter and Schmidhuber, 1997). After propagating a complete argument move through the LSTM network, the resulting hidden state is the feature vector used as input to a softmax layer which outputs the predicted label. Recurrent neural networks have also been used in the context of argument mining (Daxenberger et al., 2017; Niculae et al., 2017). We set the size of the hidden state to 75 based on several factors. Following Bengio (Bengio, 2012), we decided to have an overcomplete network, i.e. one in which the size of the hidden state is bigger than the size of the input. Since the dimensionality of our character-based encoding is 37 and that for word-based embeddings is 50, we chose a hidden state with size greater than 50 (we use the same hidden state size for both models). Increasing the size introduced overfitting even quicker than the CNN model, given that the number of weights increases more quickly for our LSTM model.

When using text as input to a neural network, we can generally view an argument move as either a sequence of characters, or as a sequence of words. Unlike previous neural network-based argument mining models, each of our models was evaluated under both conditions: for character-based models we used a one-hot encoding (one-out of n) for each letter and number - special characters were filtered since they don't hold particular meaning in speech, and we cannot be sure of transcription conventions; for word-based models we used Global Vectors (GloVe) (Pennington et al., 2014) with dimensionality of 50. An important aspect to consider is that, while word-based models have some prior knowledge encoded in the word embeddings, character-based models do not.

Since neural network models usually require a large amount of training data to be effective, and we have relatively fewer number of argument moves compared to number of model weights, we also tested hybrid models in which a neural network output is combined with handcrafted features before the final softmax classification layer, as shown in Figure 1 (b) and Figure 1 (d). Both CNN and LSTM models used categorical cross-entropy as loss function, and the number of epochs was automatically selected at training time by monitoring performance on a validation set consisting of 10% of the training set for each fold.

4.3 Multi-task Learning

As we can see from Figure 2, the argument label distributions are different for the three specificity levels. This leads us to believe a relationship exists between the specificity and argumentation annotations, therefore we decided to see whether specificity labels can be used to improve the performance of our argument mining models.

Multi-task learning for neural network models has shown promising results in the machine learning field (Weston et al., 2012; Andrychowicz et al., 2016). It has recently been used in argument mining: Schulz et al. (2018) proposed a multi-task learning setup in which the primary task consists of jointly performing argument component identification and classification (framed as a sequence tagging problem), while the additional tasks consist of the same task applied to different datasets. They showed that the multi-task models achieved better performance than single-task learning especially when limited in-domain training data is available for the primary task.

Unlike (Schulz et al., 2018), we decided to implement as secondary task specificity prediction on the same data as the primary task. The underlying neural network setup was also different: while Schulz et al. used a bidirectional LSTM followed by a Conditional Random Field (CRF) classifier (Reimers and Gurevych, 2017), we were restricted to non-sequence classifiers. We implemented multi-task learning in one of the standard ways: the embeddings generated by the networks are completely shared for both tasks of predicting argumentation and specificity. For the CNN model, we added a second softmax layer for predicting specificity after the convolutional/pooling layers. Similarly, for the LSTM model we added
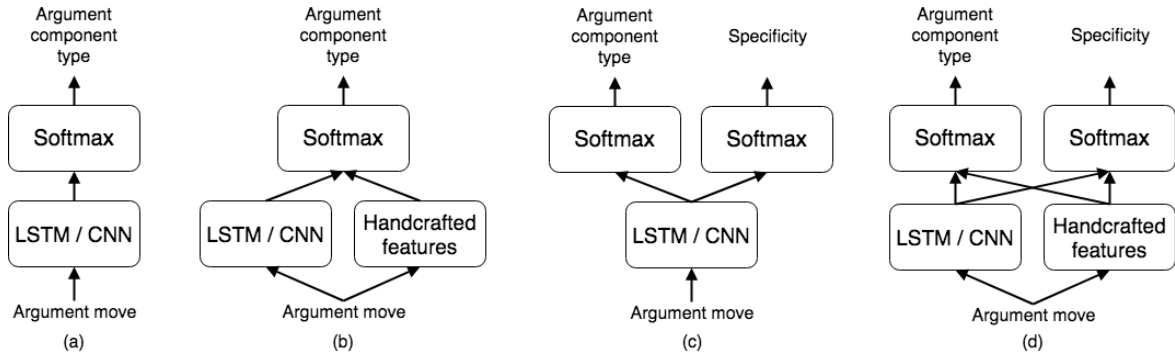
Figure 1: Neural network models used in this study: neural network only setup (a); model incorporating neural network and handcrafted features (wLDA and online dialogue sets) (b); multi-task setup for neural network only model (c); multi-task setup for model using neural network and handcrafted features (d).
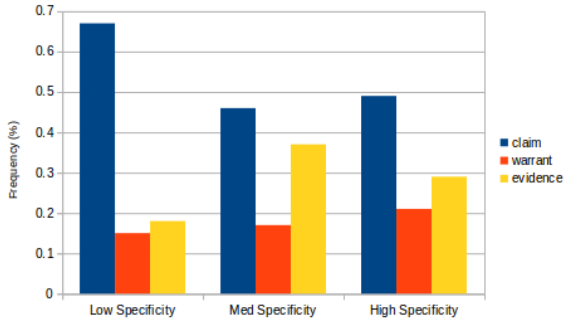

Figure 2: Argument labels by specificity levels.

a second softmax layer that operates on the final hidden state of the network to predict specificity. In both multi-task models specificity and argumentation are predicted at the same time, the loss function is computed as the sum of the individual loss functions for both tasks (the loss function for the specificity softmax layer is categorical cross-entropy as well), and gradient updates are backpropagated through the network. This process results in embeddings trained jointly for the two tasks, which can effectively capture information relevant to both specificity and argumentation.

4.4 Online Dialogue Features

Since our dataset is based on multi-party discussion, it shares similarities with prior argumentation work in multi-party online dialogues. Therefore we experiment with features from (Swanson et al., 2015), organized into three main subsets: semantic-density features (number of pronouns, descriptive word-level statistics, number of occurrences of words of different lengths), lexical features (tf-idf feature for each unigram and bigram, descriptive argument move-level statistics), and syntactic features (unigrams, bigrams and trigrams of part of speech tags). The only difference between the original features and the ones

we implemented consists in the use of Speciteller (Li and Nenkova, 2015). As observed by Lugini and Litman (Lugini and Litman, 2017), applying Speciteller as-is to domains other than news articles results in a considerable drop in performance. Therefore, instead of including the specificity score obtained by directly applying Speciteller to an argument move, we decided to use Speciteller's features.

5 Experiments and Results

This section provides our experimental results. In Section 5.1 we will test our first hypothesis: using an argument mining system trained in a different domain will result in low performance, which can be improved by re-training on classroom discussions and by adding new features. Section 5.2 will be used to test our second hypothesis: neural network models can automatically extract important features for argument component classification. Our third hypothesis will be tested in Section 5.3: adding handcrafted features (i.e. online dialogue features, wLDA features) to the ones automatically extracted by neural networks will result in an increase of performance. Lastly, we will test our fourth hypothesis in Section 5.4: jointly learning to predict argument component type and specificity will result in more robust models and achieve a further performance improvement.

Our experiments evaluate every model using a leave-one-transcript-out cross validation: each fold contains one transcript as test set and the remaining 72 as training set. Cohen kappa, and unweighted precision, recall, and f-score were used as evaluation metrics.

The following python libraries were used for implementing and testing the different models: Scikit-learn (Pedregosa et al., 2011), Tensorflow

62

(Abadi et al., 2015), Keras (Chollet et al., 2015), NLTK (Bird et al., 2009).

Given that in our dataset warrants appear much less frequently than claims and evidence, data imbalance is a problem we need to address. If trained naively, the limited amount of training data and the unbalanced class distribution lead the neural network models to specialize towards claims and evidence, with much weaker performance on warrants. This is also the case for non neural network models, although the impact on performance is lower. To combat this phenomenon we decided to use oversampling (Buda et al., 2017) in order to create a balanced dataset, hoping to further reduce the performance gap between the different classes [3]. After computing the class frequency distribution on the training set, we randomly sampled moves from the two minority classes and added them to the current training set, repeating the process until the class distribution was completely balanced (i.e. until the number of argument moves for each class equals the number of moves in the majority class) [4], while the test set was unchanged.

Table 3 shows the results for all experiments. The statistical significance results in the table use the system in row 3 as the comparison baseline, as wLDA represents a system specifically designed for argument component classification (among other tasks). Additional statistical comparisons are provided in the text as well.

5.1 Using wLDA Off the Shelf

Since not all the argument moves were considered when computing results for the pre-trained out of the box wLDA model (see Section 4.1), the results in row 2 are not directly comparable to others. Nonetheless they show the upper bound in performance of the pre-trained model, and we can see that it is comparable to a majority baseline which always predicts the majority class in each fold. This result shows that claims and evidence expressed in written essays and classroom discussions have very little in common. This is clearer when we look at improvement obtained training a logistic regression model[5] using the same wLDA

---

[3] We also tried setting class weights at training to influence the loss function, though it only improved results marginally.

[4] In the multi-task models oversampling was carried out only with respect to argument component labels since that is the primary task.

[5] We also experimented with random forest, naive Bayes and support vector machines, but they provided inferior results compared to logistic regression.

features on our dataset (row 3), which outperforms the pre-trained wLDA in all metrics (row 2), and indicates that the wLDA features are still able to somewhat distinguish between claims and evidence while performing considerably worse on warrants. Additionally, if we add to this model the online dialogue feature set, the resulting model improves all results and obtains the best kappa overall (row 4). This confirms our hypothesis: given the similarity that exists between our domain and online dialogues, features developed for analyzing argumentation in online dialogues are also useful in classroom discussions.

5.2 Neural Network Models Alone

Our second hypothesis is validated by the results in Table 3 by comparing row 3 with rows 7, 11, 15, and 19, where we can see that the CNN models achieve performance comparable to a classifier trained on features specifically developed for argument component classification. This indicates that convolutional neural network models are able to extract useful features. Additionally, when comparing the best of these models (row 19, with respect to f-score) to the best performing model based only on handcrafted features (row 4), the difference in performance is not statistically significant for any of the metrics in Table 3.

Looking more closely at the results obtained using neural network models alone we can see two different trends. While LSTM models show performance comparable to random chance (e.g. row 5, with kappa close to zero and lower than the majority baseline), three of our four CNN models (rows 7, 15, 19) perform as well as or better than the wLDA based model (row 3) (except for precision in row 19 and F_e in row 7). Overall, under the same conditions CNN models almost always outperform LSTM models. One interesting difference between the two models is that the prior knowledge introduced by word embeddings in word-based models is essential for improving performance of LSTMs (e.g. row 5 vs row 9), while this is not the case for CNN models (e.g. row 7 vs row 11). The length of sequences (i.e. argument moves) for character-based models makes it extremely hard for LSTMs to capture long-term dependencies, especially with limited amount of training data. Convolutional models, on the other hand, learn kernels that effectively function as feature detectors and seem to be able to better distinguish important features, and do not always bene-

Row	Models / Features	Kappa	Precision	Recall	F-score	F_e	F_w	F_c
1	Majority baseline	0.068	0.265	0.406	0.314	0.109	0.004	0.532
2	Pre-trained wLDA	0.077	0.289	0.350	0.269	0.351	N/A	0.456
3	Logistic Regression (wLDA features)	0.142	0.412	0.394	0.379	0.390	0.211	0.540
4	Logistic Regression (wLDA features + online dialogue)	**0.283**	0.508	0.500	0.480	0.479	0.222	**0.693**
Character level NN models								
5	LSTM	-0.002	0.062	0.253	0.082	0.007	0.242	0.013
6	LSTM + wLDA + on-line dialogue	0.034	0.217	0.304	0.150	0.080	0.272‡	0.090
7	CNN	0.143	0.439	0.423	0.393	0.372	0.218	0.574
8	CNN + wLDA + on-line dialogue	0.241*	0.482	0.475	0.450	0.449	0.236	0.637
Word level NN models								
9	LSTM	0.069	0.408	0.399	0.218	0.161	0.198	0.295
10	LSTM + wLDA + on-line dialogue	0.181	0.462	0.447	0.391	0.362	0.279‡	0.522
11	CNN	0.125	0.410	0.404	0.378	0.370	0.231	0.526
12	CNN + wLDA + on-line dialogue	0.241*	0.492*	0.488	0.455†	0.468	0.276‡	0.622
Multi-task character level NN models								
13	LSTM	0.060	0.408	0.399	0.208	0.134	0.203	0.287
14	LSTM + wLDA + on-line dialogue	0.117	0.379	0.375	0.287	0.362	0.279‡	0.522
15	CNN	0.166	0.444	0.437	0.407	0.399	0.220	0.586
16	CNN + wLDA + on-line dialogue	0.259†	0.506†	0.488	0.468*	0.474	0.262†	0.640
Multi-task word level NN models								
17	LSTM	0.093	0.379	0.364	0.276	0.298	0.252	0.378
18	LSTM + wLDA + on-line dialogue	0.232	0.497†	0.482	0.440	0.419	0.299‡	0.583
19	CNN	0.164	0.351	0.443	0.441	0.476	0.249	0.598
20	CNN + wLDA + on-line dialogue	0.276‡	**0.521**‡	**0.512**†	**0.485**†	**0.484**	**0.312**‡	0.638

Table 3: Results obtained with the baseline model/features and the proposed neural network models using different feature sets. Each line represents the average of a transcript-wise cross validation. Best results are in bold. *, †, and ‡ indicate statistical significance at the 0.1, 0.05, and 0.01 levels respectively, compared to the model in row 3. The three right-most columns represent per-class F-score for evidence, warrants, and claims respectively.

fit from word level inputs.

5.3 Adding wLDA Features and Online Dialogue Features

It is clear from Table 3 that almost all neural network models benefit from additional handcrafted features (with the exception of precision and recall for rows 13 and 14). This is not surprising, given that neural networks require a large amount of data to be trained effectively, and although random oversampling helped, we still have a limited amount of training data. Even when including additional features the two architectures show slightly different trends: CNN usually outperform LSTM, however LSTM models benefit more from the additional features. This is at least in part due to LSTMs initially having lower performance without handcrafted features. We analyzed the importance of different subsets of the online dialogue features through a feature ablation study. For CNN models, removing any subset of features resulted in a decrease in performance, except for the *syntax* subset in the *character level CNN + wLDA + online dialogue* model in both single task and multi-task settings. For LSTM models, all feature subsets contributed to increasing performance in the multi-task settings, while that was not always true for the single task models.

5.4 Multi-task Learning

Finally, we analyze the impact of multi-task learning in argument component classification. Our findings are in line with the literature in other domains, with results showing that models trained on argumentation and specificity labels almost always outperform the ones trained only on argumentation. LSTMs benefit from the multi-task setup more than CNN models: among all combinations of LSTM models, the only one able to achieve kappa greater than 0.2 and f-score greater than 0.4 is a multi-task one. Additionally, the word-level CNN model using wLDA and online dialogue feature sets and trained using multi-task learning is the only model able to achieve f-score greater than 0.3 for warrants.

It should be noted that although the neural network based model at row 20 outperforms the logistic regression model at row 4 in terms of precision, recall, and F-score, the difference in performance is not statistically significant, and neither is the reduction in kappa and F_c.

6 Conclusions and Future Work

In this work we evaluated the performance of an existing argument mining system developed for a different educational application (i.e. student essays) on a corpus composed of spoken classroom discussions. Although the pre-trained system showed poor performance on our dataset, its features show promising results when used in a model specifically trained on classroom discussions. We extracted additional feature sets based on related work in the online dialogue domain, and showed that combining online dialogue and student essay features achieves the highest kappa on our dataset. We then developed additional models based on two types of neural networks, showing that performance can be further improved. We provided an experimental evaluation of the differences between convolutional networks and recurrent networks, and between character-based and word-based models. Lastly, we showed that argument component classification models can benefit from multi-task learning, when adding a secondary task consisting of predicting specificity.

Even though we were able to achieve better performance compared to a pre-trained system and a majority baseline, we are far from the performance of argument mining systems in other domains such as student essays or legal texts. Although the wLDA features extract information from previous argument moves, we plan to take advantage of the collaborative nature of our corpus by extending the feature sets in order to exploit contextual information and develop models that can explicitly take advantage of previous argument moves. Given the performance improvements obtained with multi-task models, we also plan to extend these models and include additional tasks at training time with the hope of further boosting performance. We also plan to add other types of cross validation, since leave-one-transcript-out introduces great variability in the composition of test sets, possibly attenuating the statistical significance for some results.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. 2016. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems*, pages 3981–3989.

Arthur N Applebee, Judith A Langer, Martin Nystrand, and Adam Gamoran. 2003. Discussion-based approaches to developing understanding: Classroom instruction and student performance in middle and high school english. *American Educational Research Journal*, 40(3):685–730.

Kevin D Ashley and Vern R Walker. 2013. Toward constructing evidence-based legal arguments using legal decision documents and machine learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Law*, pages 176–180. ACM.

Yoshua Bengio. 2012. Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade*, pages 437–478. Springer.

Douglas Biber. 1988. *Variation across speech and writing*. Cambridge University Press.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. 2017. A systematic study of the class imbalance problem in convolutional neural networks. *arXiv preprint arXiv:1710.05381*.

W. Chafe and D. Tannen. 1987. The relation between written and spoken language. *Annual Review of Anthropology*, 16(1):383–407.

James S Chisholm and Amanda J Godley. 2011. Learning about language through inquiry-based discussion: Three bidialectal high school students talk about dialect variation, identity, and power. *Journal of Literacy Research*, 43(4):430–468.

François Chollet et al. 2015. Keras. https://keras.io.

Johannes Daxenberger, Steffen Eger, Ivan Habernal, Christian Stab, and Iryna Gurevych. 2017. What is the essence of a claim? cross-domain claim identification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2055–2066. Association for Computational Linguistics.

Tracy Elizabeth, Trisha L Ross Anderson, Elana H Snow, and Robert L Selman. 2012. Academic discussions: An analysis of instructional discourse and an argument for an integrative assessment framework. *American Educational Research Journal*, 49(6):1214–1250.

Randi A Engle and Faith R Conant. 2002. Guiding principles for fostering productive disciplinary engagement: Explaining an emergent argument in a community of learners classroom. *Cognition and Instruction*, 20(4):399–483.

Debanjan Ghosh, Smaranda Muresan, Nina Wacholder, Mark Aakhus, and Matthew Mitsui. 2014. Analyzing argumentative discourse units in online interactions. In *Proceedings of the First Workshop on Argumentation Mining*, pages 39–48.

Ivan Habernal and Iryna Gurevych. 2017. Argumentation mining in user-generated web discourse. *Computational Linguistics*, 43(1):125–179.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Klaus Krippendorff. 2004. Measuring the reliability of qualitative text analysis data. *Quality and Quantity*, 38:787–800.

John Lawrence and Chris Reed. 2017. Using complex argumentative interactions to reconstruct the argumentative structure of large-scale debates. In *Proceedings of the 4th Workshop on Argument Mining*, pages 108–117.

Junyi Jessy Li and Ani Nenkova. 2015. Fast and accurate prediction of sentence specificity. In *Proceedings of the Twenty-Ninth Conference on Artificial Intelligence (AAAI)*, pages 2281–2287.

Luca Lugini and Diane Litman. 2017. Predicting specificity in classroom discussion. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–61.

Luca Lugini, Diane Litman, Godley Amanda, and Olshefski Christopher. 2018. Annotating stdent talk in text-based classroom discussions. In *Proceedings*

of the 13th Workshop on Innovative Use of NLP for Building Educational Applications, pages 110–116.

Bruce M McLaren, Oliver Scheuer, and Jan Mikšátko. 2010. Supporting collaborative learning and e-discussions using artificial intelligence techniques. *International Journal of Artificial Intelligence in Education*, 20(1):1–46.

Amita Misra, Pranav Anand, Jean Fox Tree, and Marilyn Walker. 2015. Using summarization to discover argument facets in dialog. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

P Karen Murphy, Ian AG Wilkinson, Anna O Soter, Maeghan N Hennessey, and John F Alexander. 2009. Examining the effects of classroom discussion on students comprehension of text: A meta-analysis. *Journal of Educational Psychology*, 101(3):740.

Huy Nguyen and Diane Litman. 2015. Extracting argument and domain words for identifying argument components in texts. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 22–28.

Huy Nguyen and Diane J Litman. 2016. Improving argument mining in student essays by learning and exploiting argument indicators versus essay topics. In *FLAIRS Conference*, pages 485–490.

Huy V Nguyen and Diane J Litman. 2018. Argument mining for improving the automated scoring of persuasive essays. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*.

Vlad Niculae, Joonsuk Park, and Claire Cardie. 2017. Argument Mining with Structured SVMs and RNNs. In *Proceedings of ACL*.

Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 98–107. ACM.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Andreas Peldszus and Manfred Stede. 2013. From argument diagrams to argumentation mining in texts: A survey. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)*, 7(1):1–31.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Isaac Persing and Vincent Ng. 2015. Modeling argument strength in student essays. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 543–552.

Nils Reimers and Iryna Gurevych. 2017. Argumentation mining in user-generated web discourse. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348.

Alina Reznitskaya and Maughn Gregory. 2013. Student thought and classroom language: Examining the mechanisms of change in dialogic teaching. *Educational Psychologist*, 48(2):114–133.

Claudia Schulz, Steffen Eger, Johannes Daxenberger, Tobias Kahse, and Iryna Gurevych. 2018. Multi-task learning for argumentation mining in low-resource settings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 35–41. Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2014. Annotating argument components and relations in persuasive essays. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1501–1510.

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.

Reid Swanson, Brian Ecker, and Marilyn Walker. 2015. Argument mining: Extracting arguments from online dialogue. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 217–226.

Stephen Toulmin. 1958. *The uses of argument*. Cambridge: Cambridge University Press.

Jason Weston, Frédéric Ratle, Hossein Mobahi, and Ronan Collobert. 2012. Deep learning via semi-supervised embedding. In *Neural Networks: Tricks of the Trade*, pages 639–655. Springer.