Analyzing Argumentative Discourse Units in Online Interactions

Debanjan Ghosh* Smaranda Muresan† Nina Wacholder* Mark Aakhus* Matthew Mitsui**
*School of Communication and Information, Rutgers University
†Center of Computational Learning Systems, Columbia University
**Department of Computer Science, Rutgers University
debanjan.ghosh|ninwac|aakhus|mmitsui@rutgers.edu, smara@ccls.columbia.edu

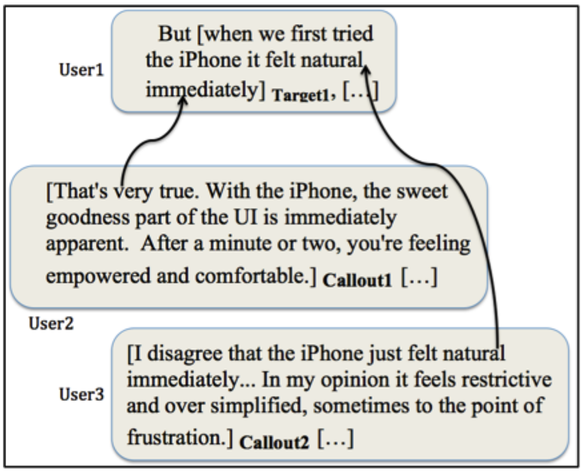
Abstract

Argument mining of online interactions is in its infancy. One reason is the lack of annotated corpora in this genre. To make progress, we need to develop a principled and scalable way of determining which portions of texts are argumentative and what is the nature of argumentation. We propose a two-tiered approach to achieve this goal and report on several initial studies to assess its potential.

1 Introduction

An increasing portion of information and opinion exchange occurs in online interactions such as discussion forums, blogs, and webpage comments. This type of user-generated conversational data provides a wealth of naturally occurring arguments. Argument mining of online interactions, however, is still in its infancy (Abbott et al., 2011; Biran and Rambow, 2011; Yin et al., 2012; Andreas et al., 2012; Misra and Walker, 2013). One reason is the lack of annotated corpora in this genre. To make progress, we need to develop a principled and scalable way of determining which portions of texts are argumentative and what is the nature of argumentation.

We propose a multi-step coding approach grounded in findings from argumentation research on managing the difficulties of coding arguments (Meyers and Brashers, 2010). In the first step, trained expert annotators identify basic argumentative features (coarse-grained analysis) in full-length threads. In the second step, we explore the feasibility of using crowdsourcing and novice annotators to identify finer details and nuances of the basic argumentative units focusing on limited thread context. Our coarse-grained scheme for argumentation is based on Pragmatic Argumentation Theory (PAT) (Van Eemeren et al., 1993; Hutchby, 2013; Maynard, 1985). PAT states that an argument can arise at any point when two or more actors engage in *calling out* and making problematic some aspect of another actor's *prior contribution* for what it (could have) said or meant (Van Eemeren et al., 1993). The argumentative relationships among contributions to a discussion are indicated through links between what is *targeted* and how it is *called-out*. Figure 1 shows an example of two Callouts that refer back to the same Target.



But [when we first tried the iPhone it felt natural immediately] Target1, [...]

[That's very true. With the iPhone, the sweet goodness part of the UI is immediately apparent. After a minute or two, you're feeling empowered and comfortable.] Callout1 [...]

[I disagree that the iPhone just felt natural immediately... In my opinion it feels restrictive and over simplified, sometimes to the point of frustration.] Callout2 [...]

Figure 1: Argumentative annotation of an Online Thread

The annotation task performed by the trained annotators includes three subtasks that Peldszus and Stede (2013a) identify as part of the argument mining problem: 1) Segmentation, 2) Segment classification, and 3) Relationship identification. In the language of Peldszus and Stede (2013a), Callouts and Targets are the basic Argument Discourse Units (ADUs) that are segmented, classified, and linked. There are two key advantages of our coarse-grained annotation scheme: 1) It does not initially prescribe what constitutes an argumentative text; 2) It makes it possible for Expert Annotators (EAs) to find ADUs in long

Proceedings of the First Workshop on Argumentation Mining, pages 39–48,
Baltimore, Maryland USA, June 26, 2014. ©2014 Association for Computational Linguistics

threads. Assigning finer grained (more complex) labels would have unduly increased the already heavy cognitive load for the EAs. In Section 2 we present the corpus, describe the annotation scheme and task, calculate Inter Annotator Agreement (IAA), and propose a hierarchical clustering approach to identify text segments that the EAs found easier or harder to annotate.

In Section 3, we report on two Amazon Mechanical Turk (MTurk) experiments, which demonstrate that crowdsourcing is a feasible way to obtain finer grained annotations of basic ADUs, especially on the text segments that were easier for the EAs to code. In the first crowd sourcing study, the Turkers (the workers at MTurk, who we consider novice annotators) assigned labels (Agree/Disagree/Other) to the relations between Callout and Target identified by the EAs. In the second study, Turkers labeled segments of Callouts as Stance or Rationale. Turkers saw only a limited context of the threaded discussion, i.e. a particular Callout-Target pair identified by the EA(s) who had analyzed the entire thread. In addition we report on initial classification experiments to detect agreement/disagreement, with the best F1 of 66.9% for the Agree class and 62.6% for the Disagree class.

2 Expert Annotation for Coarse-Grained Argumentation

Within Pragmatic Argumentation Theory, argumentation refers to the ways in which people (seek to) make some prior action or antecedent event disputable by performing challenges, contradictions, negations, accusations, resistance, and other behaviors that *call out* a *'Target'*, a prior action or event. In this section, we present the corpus, the annotation scheme based on PAT and the annotation task, the inter-annotator agreement, and a method to identify which pieces of text are easier or harder to annotate using a hierarchical clustering approach.

2.1 Corpus

Our corpus consists of blog comments posted as responses to four blog postings selected from a dataset crawled from Technorati between 2008-2010 [1]. We selected blog postings in the general topic of technology and considered only postings that had more than 200 comments. For the annotation we selected the first one hundred comments on each blog together with the original posting. Each blog together with its comments constitutes a thread. The topics of each thread are: *Android* (comparison of features of iPhone and Android phones), *iPad* (the usefulness of iPads), *Twitter* (the usefulness of Twitter as a microblogging platform), and *Layoffs* (downsizing and outsourcing efforts of technology companies). We refer to these threads as the *argumentative corpus*. We plan to make the corpus available to the research community.

2.2 Annotation Scheme and Expert Annotation Task

The coarse-grained annotation scheme for argumentation is based on the concept of Callout and Target of Pragmatic Argumentation Theory. The experts' annotation task was to identify expressions of Callout and their Targets while also indicating the links between them. We prepared a set of guidelines with careful definitions of all technical terms. The following is an abbreviated excerpt from the guidelines:

- **Callout**: A *Callout* is a subsequent action that selects (i.e., refers back to) all or some part of a prior action (i.e., Target) and comments on it in some way. In addition to referring back to the Target, a Callout explicitly includes either one or both of the following: Stance (indication of attitude or position relative to the Target) and Rationale (argument/justification/explanation of the Stance taken).

- **Target**: A *Target* is a part of a prior action that has been called out by a subsequent action.

Fig. 1 shows two examples of Callouts from two comments referring back to the same Target. Annotators were instructed to mark any text segment (from words to entire comments) that satisfied the definitions above. A single text segment could be a Target and a Callout. To per-form the expert annotation, we hired five graduate students who had a strong background in humanities and who received extensive training for the task. The EAs performed three annotation subtasks mentioned by Peldszus and Stede (2013a): Segmentation (identify the Argumentative Dis-course

[1] http://technorati.com/blogs/directory/

Thread	A1	A2	A3	A4	A5
Android	73	99	97	118	110
iPad	68	86	85	109	118
Layoffs	71	83	74	109	117
Twitter	76	102	70	113	119
Avg.	72	92.5	81.5	112.3	116

Table 1: Number of Callouts by threads and EA

Thread	F1_EM	F1_OM	α
Android	54.4	87.8	0.64
iPad	51.2	86.0	0.73
Layoffs	51.9	87.5	0.87
Twitter	53.8	88.5	0.82

Table 2: IAA for 5 EA: F1 and alpha values per thread

Units (ADUs) including their boundaries), Segment classification (label the roles of the ADUs, in this case Callout and Target) and relation identification (indicate the link between a Callout and the most recent Target to which is a response).

The segmentation task, which Artstein and Poesio (2008) refer to as the unitization problem, is particularly challenging. Table 1 shows extensive variation in the number of ADUs (Callout in this case) identified by the EAs for each of the four threads. Annotator A1 identified the fewest Callouts (72) while A4 and A5 identified the most (112.3 and 116, respectively). Although these differences could be due to the issues with training, we interpret the consistent variation among coders as an indication that judges can be characterized as "lumpers" or "splitters". What lumpers considered a single long unit was treated as two (or more) shorter units by splitters. This is an example of the problem of annotator variability discussed in (Peldszus and Stede, 2013b). Similar behavior was noticed for Targets. [2]

2.3 Inter Annotator Agreement

Since the annotation task includes the segmentation step, to measure the IAA we have to account for fuzzy boundaries. Thus, we con-sider two IAA metrics usually used in literature for such cases: the information retrieval (IR) in-spired precision-recall (P/R/F1) measure (Wiebe et al., 2005) and Krippendorff's α (Krippendorff, 2004). We present here the main results; a detailed discussion of the IAA is left for a different paper. Following Wiebe et al. (2005), to calculate P/R/F1 for two annotators, one annotator's ADUs are selected as the gold standard. If more than two annotators are employed, the IAA is the average of the pairwise P/R/F1. To determine if two annotators have selected the same text span to represent an ADU, we use the two methods of Somasundaran et al. (2008): exact match (EM) - text spans that vary at the start or end point by five characters or less, and overlap match (OM) - text spans that have at least 10% of same overlapping characters. Table 2 shows the F1 measure for EM and OM for the five EAs on each of the four threads. As expected, the F1 measures are much lower for EM than for OM.

For the second IAA metric, we implement Krippendorff's α (Krippendorff, 2004), where the character overlap between any two annotations and the gap between them are utilized to measure the expected disagreement and the observed disagreement. Table 2 shows α values for each thread, which means significant agreement.

While the above metrics show reasonable agreement across annotators, they do not tell us what pieces of text are easier or harder to annotate. In the next section we report on a hierarchical clustering technique that makes it possible to assess how difficult it is to identify individual text segments as Callouts.

2.4 Clustering of Callout ADUs

We use a hierarchical clustering technique (Hastie et al., 2009) to cluster ADUs that are variants of the same Callout. Each ADU starts in its own cluster. The start and end points of each ADU are utilized to identify overlapping characters in pairs of ADUs. Then, using a "bottom up" clustering approach, two ADUs (in this case, pairs of Callouts) that share overlapping characters are merged into a cluster. This process continues until no more text segments can be merged. Clusters with five overlapping ADUs include a text segment that all five annotators have labeled as a Callout, while clusters with one ADU indicates that only one annotator classified the text segment as a Callout (see Table 3). These numbers provide information about what segments of text are easier or harder to code. For instance, when a cluster contains only two ADUs, it means that three of the five annotators did not label the text segment as a Callout. Our MTurk study of Stance/Rationale (Sec. 3.2) could highlight one reason for the variation – some coders consider a segment of text as Callout when an implicit Stance is present, while others do not.

[2] Due to space limitations, here and in the rest of this paper we report only on Callouts.

# Of EAs	Callout	Target
5	I disagree too. some things they get right, some things they do not.	the iPhone is a truly great design.
	I disagree too … they do not.	That happened because the iPhone is a truly great design.
	I disagree too.	But when we first tried the iPhone it felt natural immediately … iPhone is a truly great design.
	Hi there, I disagree too … they do not. Same as OSX.	–Same as above-
	I disagree too… Same as OSX … no problem.	–Same as above-
2	Like the reviewer said … (Apple) the industry leader.… Good luck with that (iPhone clones).	Many of these iPhone … griping about issues that will only affect them once in a blue moon
	Like the reviewer said… (Apple) the industry leader.	Many of these iPhone…
1	Do you know why the Pre … various handset/builds/resolution issues?	Except for games?? iPhone is clearly dominant there.

Table 3: Examples of Callouts lusters and their corresponding Targets

Thread	# of Clusters	# of EA ADUs per cluster				
		5	4	3	2	1
Android	91	52	16	11	7	5
Ipad	88	41	17	7	13	10
Layoffs	86	41	18	11	6	10
Twitter	84	44	17	14	4	5

Table 4: Number of clusters for each cluster type

Table 4 shows the number of Callout clusters in each thread. The number of clusters with five and four annotators shows that in each thread there are Callouts that are plausibly easier to identify. On the other hand, the clusters selected by only one or two annotators are harder to identify.

3 Crowdsourcing for Fine-grained Argumentation

To understand better the nature of the ADUs, we conducted two studies asking Turkers to perform finer grained analysis of Callouts and Targets. Our first study asked five Turkers to label the relation between a Callout and its corresponding Target as Agree, Disagree, or Other. The Other relation may be selected in a situation where the Callout has no relationship with the Target (e.g., a possible digression) or is in a type of argumentative relationship that is difficult to classify as either Agreement or Disagreement. The second study asked five Turkers to identify Stance and Rationale in Callouts identified by EAs. As discussed in Section 2, by definition, a Callout contains an explicit instance of Stance, Rationale or both. In both of these crowdsourcing studies the Turkers were shown only a limited portion of the threaded discussion, i.e. the Callout-Target pairs that the EAs had linked.

Crowdsourcing is becoming a popular mechanism to collect annotations and other type of data for natural language processing research (Wang and Callison-Burch, 2010; Snow et al., 2008; Chen and Dolan, 2011; Post et al., 2012). Crowdsourcing platforms such as Amazon Mechanical Turk (MTurk) provide a flexible framework to submit various types of NLP tasks where novice annotators (Turkers) can generate content (e.g., translations, paraphrases) or annotations (labeling) in an inexpensive way and with limited training. MTurk also provides researchers with the ability to control the quality of the Turkers, based on their past performances. Section 3.1 and 3.2 describe our two crowdsourcing studies for fine grain argumentation annotation.

3.1 Crowdsourcing Study 1: Labeling the Relation between Callout and Target

In this study, the Turkers' task was to assign a relation type between a Callout and its associated Target. The choices were Agree, Disagree, or Other. Turkers were provided with detailed instructions, including multiple examples of Callout and Target pairs and their relation type. Each HIT (Human Intelligence Task, in the language of MTurk) contained one Callout-Target pair and Turkers were paid 2 cents per HIT. To assure a level of quality control, only qualified Turkers were allowed to perform the task (i.e., Master level with more than 95% approval rate and at least 500 approved HITs).

For this experiment, we randomly selected a Callout from each cluster, along with its corresponding Target. Our assumption is that all Callout ADUs in a given cluster have the same relation type to their Targets (see Table 3). While this assumption is logical, we plan to fully investigate it in future work by running an MTurk experiment on all the Callout ADUs and their corresponding Targets.

We utilized Fleiss' kappa (Fleiss, 1971) to compute IAA between the Turkers (every HIT was completed by five Turkers). Kappa is between 0.45-0.55 for each thread showing moderate agreement between the Turkers (Landis et al., 1977). These agreement results are in line with the agreement noticed in previous studies on agreement/disagreement annotations in online interactions (Bender et al., 2011; Abbott et al., 2011). To select a gold standard for the relation type, we used majority voting. That is, if three or more Turkers agreed on a label, we selected that label as the gold standard. In cases where there was no majority, we assigned the label Other. The total number of Callouts that are in agreement and in disagreement with Targets are 143 and 153, respectively.

Table 5 shows the percentage of each type of relation identified by Turkers (Agree/Disagree/Other) for clusters annotated by different number of EAs. The results suggest that there is a correlation between text segments that are easier or harder to annotate by EAs with the ability of novice annotators to identify an Agree/Disagree relation type between Callout and Target. For example, Turkers generally discovered Agree/Disagree relations between Callouts and their Targets when the Callouts are part of those clusters that are annotated by a higher number of EAs. Turkers identified 57% as showing a disagreement relation between Callout and Target, and 39% as showing an agreement relation (clusters with 5 EAs). For those clusters, only 4% of the Callouts are labeled as having an Other relation with the Target. For clusters selected by fewer EAs, however, the number of Callouts having a relation with the Target labeled as Other is much higher (39% for clusters with two EAs and 32% for clusters with one EA). These results show that those Callouts that are easier to discover (i.e., identified by all five EAs) mostly have a relation with the Target (Agree or Disagree) that is clearly expressed and thus recognizable to the Turkers. Table 5 also shows that in some cases even if some EAs agreed on a piece of text to be considered as a Callout, the novice annotators assigned the Other relation to the Callout and Target ADUs. There are two possible explanations: either the novice annotators could not detect an implicit agreement or disagreement and thus they selected Other, or there are other types of relations besides Agreement and Disagreement between Callouts and their corresponding Targets. We plan to extend this study to other fine grained relation types in future work. In the next section we discuss the results of building a supervised classifier to predict the Agree or Disagree relation type between Callout/Target pairs.

Relation label	# of EA ADUs per cluster				
	5	4	3	2	1
Agree	39.36	43.33	42.50	35.48	48.39
Disagree	56.91	31.67	32.50	25.81	19.35
Other	3.72	25.00	25.00	38.71	32.26

Table 5: Percentage of Relation labels per EA cluster type

3.1.1 Predicting the Agree/Disagree Relation Label

We propose a supervised learning setup to classify the relation types of Callout-Target pairs. The classification categories are the labels collected from the MTurk experiment. We only consider the Agree and Disagree categories since the Other category has a very small number of instances (53). Based on the annotations from the Turkers, we have 143 Agree and 153 Disagree training instances.

We first conducted a simple baseline experiment to check whether participants use words or phrases to express explicit agreement or disagreement such as 'I agree', 'I disagree'. We collected two small lists (twenty words each) of words from Merriam-Webster dictionary that explicitly represent agreement and disagreement Stances. The agreement list contains the word 'agree' and its synonyms such as 'accept', 'concur', and 'accede'. The disagreement list contains the word 'disagree' and synonyms such as 'differ' and 'dissent'. We then checked whether the text of the Callouts contains these explicit agreement/disagreement markers. Note, that these markers are utilized as rules and no statistical learning is involved in this stage of experiment.

The first row of the Table 6 represents the baseline results. Though the precision is high for agreement category, the recall is quite low and that results in a poor overall F1 measure. This shows that even though markers like 'agree' or 'disagree'

Features	Category	P	R	F1
Baseline	Agree	83.3	6.9	12.9
	Disagree	50.0	5.2	9.5
Unigrams	Agree	57.9	61.5	59.7
	Disagree	61.8	58.2	59.9
MI-based unigram	Agree	60.1	66.4	63.1
	Disagree	65.2	58.8	61.9
LexF	Agree	61.4	73.4	66.9
	Disagree	69.6	56.9	62.63

Table 6: Classification of Agree/Disagree

Features	Category	P	R	F1
LexF	Agree	61.4	73.4	66.9
	Disagree	69.6	56.9	62.6
LexF-SL	Agree	60.6	74.1	66.7
	Disagree	69.4	54.9	61.3
LexF-IU	Agree	58.1	69.9	63.5
	Disagree	65.3	52.9	58.5
LexF-LO	Agree	57.2	74.8	64.8
	Disagree	67.0	47.7	55.7

Table 7: Importance of Lexical Features

are very precise, they occur in less than 15% of all the Callouts expressing agreement or disagreement.

For the next set of experiments we used a supervised machine learning approach for the two-way classification (Agree/Disagree). We use Support Vector Machines (SVM) as our machine-learning algorithm for classification as implemented in Weka (Hall et al., 2009) and ran 10-fold cross validation. As a SVM baseline, we first use all unigrams in Callout and Target as features (Table 6, Row 2). We notice that the recall improves significantly when compared with the rule-based method. To further improve the classification accuracy, we use Mutual Information (MI) to select the words in the Callouts and Targets that are likely to be associated with the categories Agree and Disagree, respectively. Specifically, we sort each word based on its MI value and then select the first 180 words in each of the two categories to represent our new vocabulary set of 360 words. The feature vector includes only words present in the MI list. Compared to the all unigrams baseline, the MI-based unigrams improve the F1 by 4% (Agree) and 2% (Disagree) (Table 6). The MI approach discovers the words that are highly associated with Agree/Disagree categories and these words turn to be useful features for classification. In addition, we consider several types of lexical features (LexF) inspired by previous work on agreement and disagreement (Galley et al., 2004; Misra and Walker, 2013).

- **Sentiment Lexicon (SL)**: Two features are designed using a sentiment lexicon (Hu and Liu, 2004) where the first feature represents the number of times the Callout and the Target contain a positive emotional word and the second feature represents the number of the negative emotional words.

- **Initial unigrams in Callout (IU)**: Instead of using all unigrams in the Callout and Target, we only select the first words from the Callout (maximum ten). The assumption is that the stance is generally expressed at the beginning of a Callout. We used the same MI-based technique to filter any sparse words.

- **Lexical Overlap and Length (LO)**: This set of features represents the lexical overlap between the Callout and the Target and the length of each ADU.

Table 6 shows that using all these types of lexical features improves the F1 score for both categories as compared to the MI-based unigram features. Table 7 shows the impact of removing each type of lexical features. From these results it seems that initial unigrams of Callout (IU) and lexical overlap (LO) are useful features: removing each of them lowers the results for both Agree/Disagree categories. In future work, we plan to explore context-based features such as the thread structure, and semantic features such as WordNet-based semantic similarity. We also hypothesize that with additional training instances the ML approaches will achieve better results.

3.2 Crowdsourcing Study 2: Analysis of Stance and Rationale

In the second study aimed at identifying the argumentative nature of the Callouts identified by the expert annotators, we focus on identifying the Stance and Rationale segments of a Callout. Since the presence of at least an explicit Stance or Rationale was part of the definition of a Callout, we selected these two argumentation categories as our finer-grained scheme for this experiment.

Given a pair of Callout and Target ADUs, five Turkers were asked to identify the Stance and Rationale segments in the Callout, including the exact boundaries of the text segments. Identifying Stance and Rationale is a difficult task and thus, we also asked Turkers to mark the level of difficulty in the identification task. We provided the

Diff	Number of EAs per cluster				
	5	4	3	2	1
VE	22.11	22.38	20.25	16.67	10.71
E	28.55	24.00	24.02	28.23	20.00
M	19.69	17.87	20.72	19.39	23.57
D	11.50	10.34	11.46	9.52	12.86
VD	7.02	5.61	4.55	4.42	6.43
TD	11.13	19.79	19.00	21.77	26.33

Table 8: Difficulty judgments by Turkers compared to number of EAs who selected a cluster

Turkers with a scale of difficulty (similar to a Likert scale), where the Turkers have to choose one of the following: *very easy* (VE), *easy* (E), *moderate* (M), *difficult* (D), *very difficult* (VD), *too difficult to code* (TD). Turkers were instructed to select the too difficult to code choice only in cases where they felt it was impossible to detect a Stance or Rationale in the Callout.

The Turkers were provided with detailed instructions including examples of Stance and Rationale annotations for multiple Callouts and only highly qualified Turkers were allowed to perform the task. Unlike the previous study, we also ran a pre-screening testing phase and only Turkers that passed the screening were allowed to complete the tasks. Because of the difficult nature of the annotation task, we paid ten cents per HIT.

For the Stance/Rationale study, we considered all the Callouts in each cluster along with the associated Targets. We selected all the Callouts from each cluster because of variability in the boundaries of ADUs, i.e., in the segmentation process. One benefit of this crowdsourcing experiment is that it helps us understand better what the variability means in terms of argumentative structure. For example, one EA might mark a text segment as a Callout only when it expresses a Stance, while another EA might mark as Callout a larger piece of text expressing both the Stance and Rationale (See examples of Clusters in Table 3). We leave this deeper analysis as future work.

Table 8 shows there is a correlation between the number of EAs who selected a cluster and the difficulty level Turkers assigned to identifying the Stance and Rationale elements of a Callout. This table shows that for more than 50% of the Callouts that are identified by 5 EAs, the Stance and Rationale can be easily identified (refer to the 'VE' and 'E' rows), where as in the case of Callouts that are identified by only 1 EA, the number is just 31%. Similarly, more than 26% of the Callouts in that same category (1 EA) were labeled as 'Too difficult to code', indicating that the Turkers could not identify either a Stance or Rationale in the Callout. These numbers are comparable to what our first crowdsourcing study showed for the Agree/Disagree/Other relation identification (Table 5). Table 9 shows results where we selected overall difficulty level by majority voting. We combined the *easy* and *very easy* categories to the category *easy (E)* and the *difficult* and *very difficult* categories to the category *difficult (D)* for a simpler presentation.

Diff	Number of EAs per cluster				
	5	4	3	2	1
E	81.04	70.76	60.98	63.64	25.00
M	7.65	7.02	17.07	6.06	25.00
D	5.91	5.85	7.32	9.09	12.50
TD	5.39	16.37	14.63	21.21	37.50

Table 9: Difficulty judgment (majority voting)

Table 9 also shows that more than 80% of the time, Turkers could easily identify Stance and/or Rationale in Callouts identified by 5 EAs, while they could perform the finer grained analysis easily only 25% of time for Callouts identified by a single EA. Only 5% of Callouts identified by all 5 EAs were considered *too difficult to code* by the Turkers (i.e., the novice annotators could not identify a Stance or a Rationale). In contrast, more than 37% of Callouts annotated only by 1 EA were considered *too difficult to code* by the novice annotators. Table 10 presents some of the examples of Stance and Rationale pairs as selected by the Turkers along with the difficulty labels.

4 Related Work

Primary tasks for argument analysis are to segment the text to identify ADUs, detect the roles of each ADUs, and to establish the relationship between the ADUs (Peldszus and Stede, 2013a). Similarly, Cohen (1987) presented a computational model of argument analysis where the structure of each argument is restricted to the claim and evidence relation. Teufel et al. (2009) introduce the argumentative zoning (AZ) idea that identifies important sections of scientific articles and later Hachey and Grover (2005) applied similar idea of AZ to summarize legal documents. Wyner et al. (2012) propose a rule-based tool that can highlight potential argumentative sections of text according to discourse cues like 'suppose' or 'therefore'. They tested their system on product reviews

Target	Callout	Stance	Rationale	Difficulty
the iPhone is a truly great design.	I disagree too. some things they get right, some things they do not.	I…too	Some things …do not	Easy
the dedicated 'Back' button	that back button is key. navigation is actually much easier on the android.	That back button is key	Navigation is…android	Moderate
It's more about the features and apps and Android seriously lacks on latter.	Just because the iPhone has a huge amount of apps, doesn't mean they're all worth having.	—	Just because the iPhone has a huge amount of apps, doesn't mean they're all worth having.	Difficult
I feel like your comments about Nexus One is too positive …	I feel like your poor grammar are to obvious to be self thought...	—	—	Too difficult to code

Table 10: Examples of Callout/Target pairs with difficulty level (majority voting)

(Canon Camera) from Amazon e-commerce site.

Relatively little attention has so far been devoted to the issue of building argumentative corpora from naturally occurring texts (Peldszus and Stede, 2013a; Feng and Hirst, 2011). However, (Reed et al., 2008; Reed and Rowe, 2004) have developed the Araucaria project that maintains an online repository of arguments (AraucariaDB), which recently has been used as research corpus for several automatic argumentation analyses (Palau and Moens, 2009; Wyner et al., 2010; Feng and Hirst, 2011). Our work contributes a new principled method for building annotated corpora for online interactions. The corpus and guidelines will also be shared with the research community.

Another line of research that is correlated with ours is recognition of agreement/disagreement (Misra and Walker, 2013; Yin et al., 2012; Abbott et al., 2011; Andreas et al., 2012; Galley et al., 2004; Hillard et al., 2003) and classification of stances (Walker et al., 2012; Somasundaran and Wiebe, 2010) in online forums. For future work, we can utilize textual features (contextual, dependency, discourse markers), relevant multiword expressions and topic modeling (Mukherjee and Liu, 2013), and thread structure (Murakami and Raymond, 2010; Agrawal et al., 2003) to improve the Agree/Disagree classification accuracy.

Recently, Cabrio and Villata (2013) proposed a new direction of argumentative analysis where the authors show how arguments are associated with Recognizing Textual Entailment (RTE) research. They utilized RTE approach to detect the relation of support/attack among arguments (entailment expresses a 'support' and contradiction expresses an 'attack') on a dataset of arguments collected from online debates (e.g., Debatepedia).

5 Conclusion and Future Work

To make progress in argument mining for online interactions, we need to develop a principled and scalable way to determine which portions of texts are argumentative and what is the nature of argumentation. We have proposed a two-tiered approach to achieve this goal. As a first step we adopted a coarse-grained annotation scheme based on Pragmatic Argumentation Theory and asked expert annotators to label entire threads using this scheme. Using a clustering technique we identified which pieces of text were easier or harder for the Expert Annotators to annotate. Then we showed that crowdsourcing is a feasible approach to obtain annotations based on a finer grained argumentation scheme, especially on text segments that were easier for the Expert Annotators to label as being argumentative. While more qualitative analysis of these results is still needed, these results are an example of the potential benefits of our multi-step coding approach.

Avenues for future research include but are not limited to: 1) analyzing the differences between the stance and rationale annotations among the novice annotators; 2) improving the classification accuracies of the Agree/Disagree classifier using more training data; 3) using syntax and semantics inspired textual features and thread structure; and 4) developing computational models to detect Stance and Rationale.

Acknowledgements

Part of this paper is based on work supported by the DARPA-DEFT program for the first two authors. The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

References

Rob Abbott, Marilyn Walker, Pranav Anand, Jean E Fox Tree, Robeson Bowmani, and Joseph King. 2011. How can you say such things?!?: Recognizing disagreement in informal political argument. In *Proceedings of the Workshop on Languages in Social Media*, pages 2–11. Association for Computational Linguistics.

Rakesh Agrawal, Sridhar Rajagopalan, Ramakrishnan Srikant, and Yirong Xu. 2003. Mining newsgroups using networks arising from social behavior. In *Proceedings of the 12th international conference on World Wide Web*, pages 529–535. ACM.

Jacob Andreas, Sara Rosenthal, and Kathleen McKeown. 2012. Annotating agreement and disagreement in threaded discussion. In *LREC*, pages 818–822.

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Emily M Bender, Jonathan T Morgan, Meghan Oxley, Mark Zachry, Brian Hutchinson, Alex Marin, Bin Zhang, and Mari Ostendorf. 2011. Annotating social acts: Authority claims and alignment moves in wikipedia talk pages. In *Proceedings of the Workshop on Languages in Social Media*, pages 48–57. Association for Computational Linguistics.

Or Biran and Owen Rambow. 2011. Identifying justifications in written dialogs by classifying text as argumentative. *International Journal of Semantic Computing*, 5(04):363–381.

Elena Cabrio and Serena Villata. 2013. A natural language bipolar argumentation approach to support users in online debate interactions. *Argument & Computation*, 4(3):209–230.

David L Chen and William B Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 190–200. Association for Computational Linguistics.

Robin Cohen. 1987. Analyzing the structure of argumentative discourse. *Computational linguistics*, 13(1-2):11–24.

Vanessa Wei Feng and Graeme Hirst. 2011. Classifying arguments by scheme. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 987–996. Association for Computational Linguistics.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Michel Galley, Kathleen McKeown, Julia Hirschberg, and Elizabeth Shriberg. 2004. Identifying agreement and disagreement in conversational speech: Use of bayesian networks to model pragmatic dependencies. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 669. Association for Computational Linguistics.

Ben Hachey and Claire Grover. 2005. Automatic legal text summarisation: experiments with summary structuring. In *Proceedings of the 10th international conference on Artificial intelligence and law*, pages 75–84. ACM.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. 2009. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.

Trevor Hastie, Robert Tibshirani, Jerome Friedman, T Hastie, J Friedman, and R Tibshirani. 2009. *The elements of statistical learning*, volume 2. Springer.

Dustin Hillard, Mari Ostendorf, and Elizabeth Shriberg. 2003. Detection of agreement vs. disagreement in meetings: Training with unlabeled data. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: companion volume of the Proceedings of HLT-NAACL 2003–short papers-Volume 2*, pages 34–36. Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM.

Ian Hutchby. 2013. *Confrontation talk: Arguments, asymmetries, and power on talk radio*. Routledge.

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

J Richard Landis, Gary G Koch, et al. 1977. The measurement of observer agreement for categorical data. *biometrics*, 33(1):159–174.

Douglas W Maynard. 1985. How children start arguments. *Language in society*, 14(01):1–29.

Renee A Meyers and Dale Brashers. 2010. Extending the conversational argument coding scheme: Argument categories, units, and coding procedures. *Communication Methods and Measures*, 4(1-2):27–45.

Amita Misra and Marilyn A Walker. 2013. Topic independent identification of agreement and disagreement in social media dialogue. In *Proceedings of the SIGDIAL 2013 Conference*, pages 41–50. Association for Computational Linguistics.

Arjun Mukherjee and Bing Liu. 2013. Discovering user interactions in ideological discussions. In *Proceedings of the 51st Annual Meeting on Association for Computational Linguistics*, pages 671–681. Citeseer.

Akiko Murakami and Rudy Raymond. 2010. Support or oppose?: classifying positions in online debates from reply activities and opinion expressions. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 869–875. Association for Computational Linguistics.

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.

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

Andreas Peldszus and Manfred Stede. 2013b. Ranking the annotators: An agreement study on argumentation structure. In *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 196–204.

Matt Post, Chris Callison-Burch, and Miles Osborne. 2012. Constructing parallel corpora for six indian languages via crowdsourcing. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 401–409. Association for Computational Linguistics.

Chris Reed and Glenn Rowe. 2004. Araucaria: Software for argument analysis, diagramming and representation. *International Journal on Artificial Intelligence Tools*, 13(04):961–979.

Chris Reed, Raquel Mochales Palau, Glenn Rowe, and Marie-Francine Moens. 2008. Language resources for studying argument. In *Proceedings of the 6th conference on language resources and evaluation-LREC 2008*, pages 91–100.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. 2008. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the conference on empirical methods in natural language processing*, pages 254–263. Association for Computational Linguistics.

Swapna Somasundaran and Janyce Wiebe. 2010. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124. Association for Computational Linguistics.

Swapna Somasundaran, Josef Ruppenhofer, and Janyce Wiebe. 2008. Discourse level opinion relations: An annotation study. In *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*, pages 129–137. Association for Computational Linguistics.

Simone Teufel, Advaith Siddharthan, and Colin Batchelor. 2009. Towards discipline-independent argumentative zoning: evidence from chemistry and computational linguistics. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3*, pages 1493–1502. Association for Computational Linguistics.

Frans H Van Eemeren, Rob Grootendorst, Sally Jackson, and Scott Jacobs. 1993. *Reconstructing argumentative discourse.* University of Alabama Press.

Marilyn A Walker, Pranav Anand, Rob Abbott, Jean E Fox Tree, Craig Martell, and Joseph King. 2012. That is your evidence?: Classifying stance in online political debate. *Decision Support Systems*, 53(4):719–729.

Rui Wang and Chris Callison-Burch. 2010. Cheap facts and counter-facts. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 163–167. Association for Computational Linguistics.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2-3):165–210.

Adam Wyner, Raquel Mochales-Palau, Marie-Francine Moens, and David Milward. 2010. Approaches to text mining arguments from legal cases. In *Semantic processing of legal texts*, pages 60–79. Springer.

Adam Wyner, Jodi Schneider, Katie Atkinson, and Trevor JM Bench-Capon. 2012. Semi-automated argumentative analysis of online product reviews. In *COMMA*, pages 43–50.

Jie Yin, Paul Thomas, Nalin Narang, and Cecile Paris. 2012. Unifying local and global agreement and disagreement classification in online debates. In *Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis*, pages 61–69. Association for Computational Linguistics.