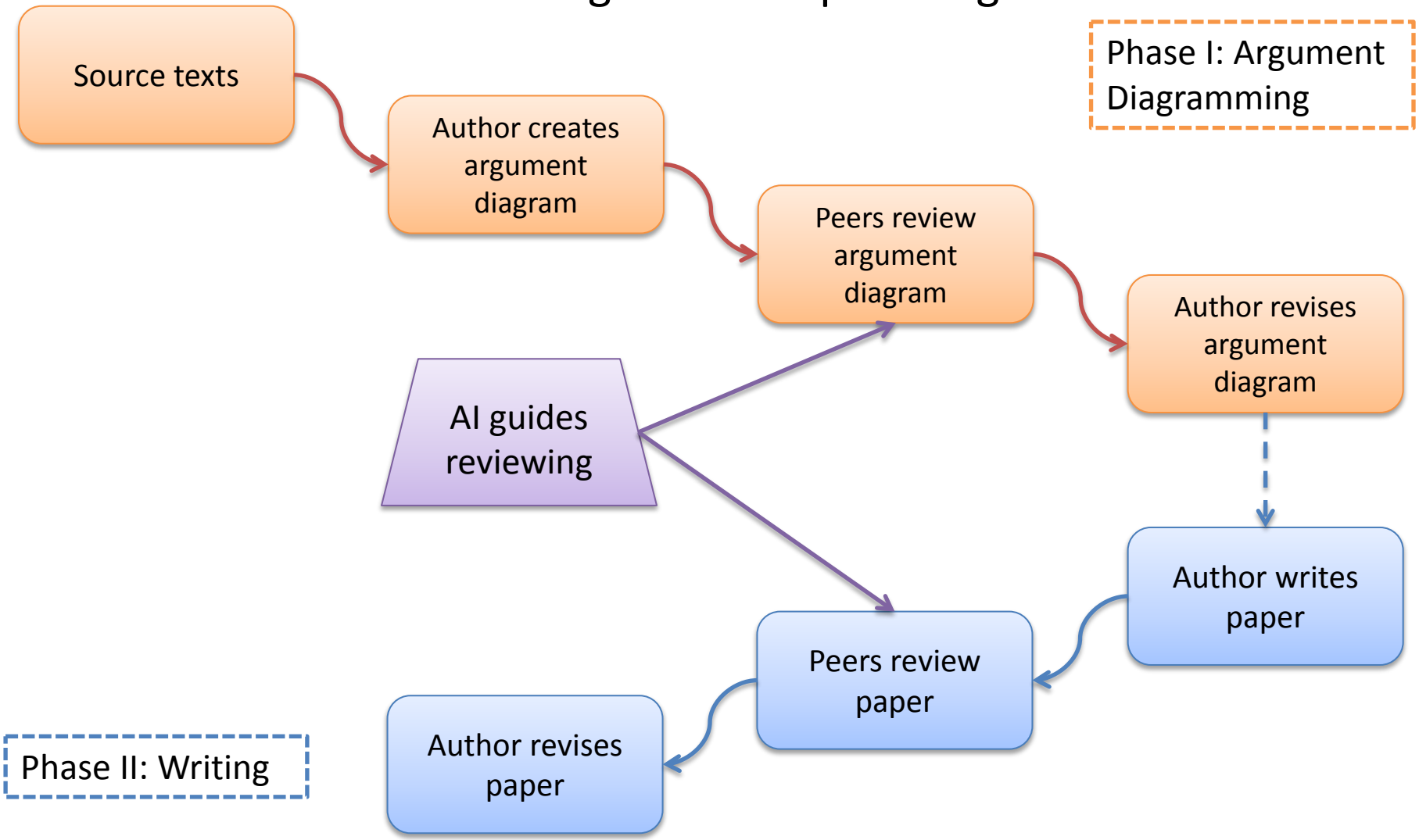
Identifying Localization in Reviews of Argument Diagrams

Huy Nguyen[1] Diane Litman[1,2]

[1]Computer Science Department
[2]Learning Research and Development Center
at University of Pittsburgh

ArgumentPeer Project*

Use diagram to help writing



* NSF Grant No. 1122504

Outline

- Introduction
- Corpus
- Location Pattern Algorithm
- Experiments
- Discussion and future work
- Integrating LPA into SWoRD

Peer reviews with SWoRD

(Cho and Schunn, 2007)

- Web-based reciprocal peer review system to facilitate writing and reviewing practices for students
- Manage typical activity cycles such as writing, reviewing, back-evaluating, and rewriting
- However, SWoRD lacks intelligence for detecting and responding to problems with student reviewing's performance

Argument diagram with LASAD

(Scheuer et. al, 2009)

- Support the learning of argumentation skills through graphical representations
- Argument diagrams with nodes represent statements and arcs represent argumentative or rhetorical relations
- By combining SWoRD and LASAD, student argument diagrams are distributed to student reviewers for comment

Review localization

- Pinpointing the source or location of a problem and/or solution (Nelson and Schunn, 2009)
- Significantly related to feedback implementation of peer paper review (Nelson and Schunn, 2009) and peer argument diagram review (Lippman et al., 2012)
- Paper review localization was proved predictable using NLP and ML techniques (Xiong and Litman, 2011)
- We address **review localization of peer argument diagram review**

Research goals

- Overall: Adapting and applying Natural Language Processing and Machine Learning techniques to help peer reviewers **review the diagram and/or writing** of others based on automatic detection of effective review comment features

- This study: Automatically identifying **review localization** in student **argument diagram reviews**

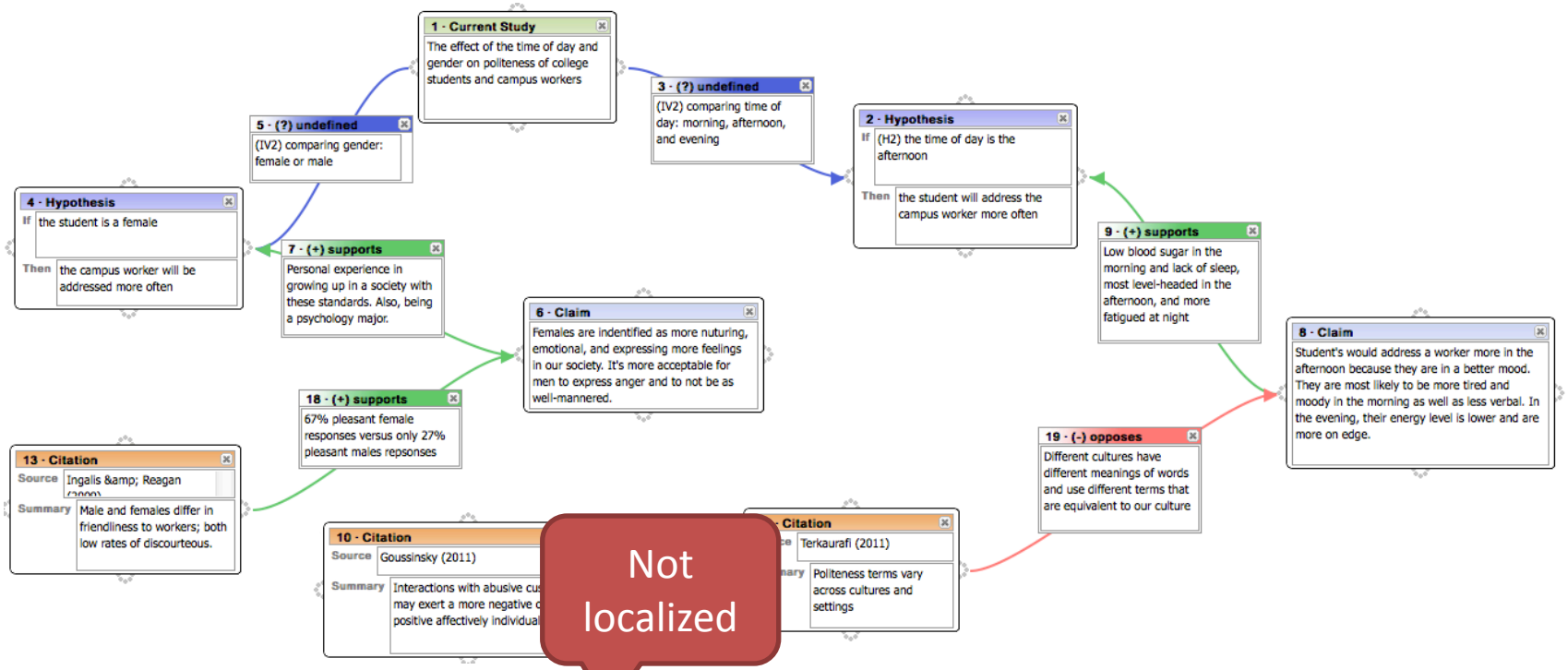
Argument diagram review corpus

- Context: Research Method Lab, Fall 2011
  - Students created argument, student reviewers then provided written feedback and grades
- Instructor-defined ontology
  - 4 node types: *current study*, *hypothesis*, *claim*, and *citation*
  - 4 arc types: *comparison*, *undefined*, *supports*, and *opposes*

- Comments were segmented into 1104 idea units (*contiguous feedback referring to a single topic*)
- On average, each diagram was reviewed by 3 peers with 19 comment units

An example argument diagram

(localization is highlighted)



The citations presented are solid evidence but are not presented in the best way possible. The justification is understandable but not convincing.

Also **the con-argument for the time of day hypothesis** is not sufficient. **Citation 15** does not oppose the claim.

Corpus

Localized

Annotation

- Two annotators coded 1104 comments for **issue types**:
  - *praise*, *summary*, *problem*, *solution*, *problem and solution* (both), or *uncodeable*
- 590 comments having types of **praise**, **problem**, or **both** were further coded for **localization** with label = {*yes*, *no*}
- Inter-rater reliability (kappa) is high:
  - 0.87 for issue type
  - 0.84 for localization

Diagram Review Localization: Observation

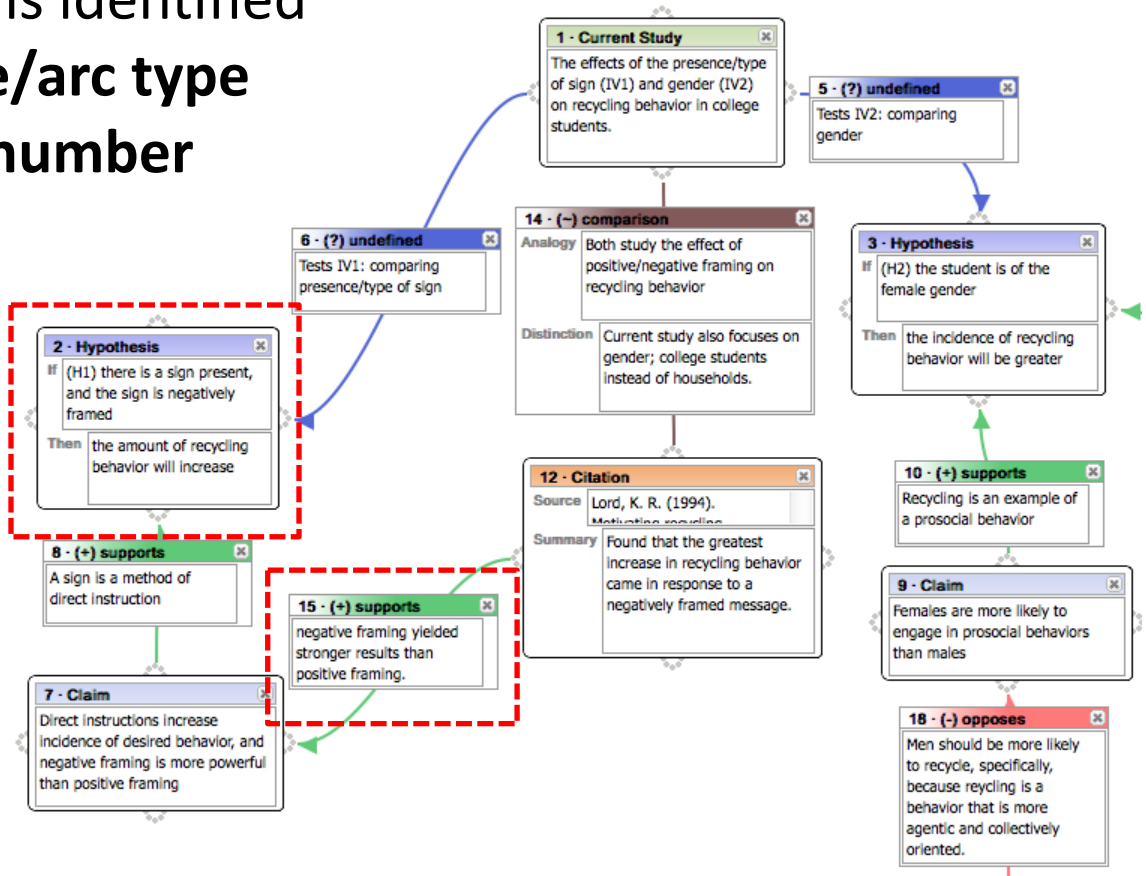
- Paper review vs. diagram review

- Graph structure of argument diagrams makes it more convenient to include location information
  - Xiong and Litman (2011) reported 53% of reviews localized
  - Our corpus has 74% of reviews localized

- The way that localization is realized in diagram review differs from that in paper review

Location Patterns

- **Numbered ontology type**
- A diagram component is identified by referring to its **node/arc type followed by ID/order number**
  - *hypothesis 1*
  - *support arc 15*

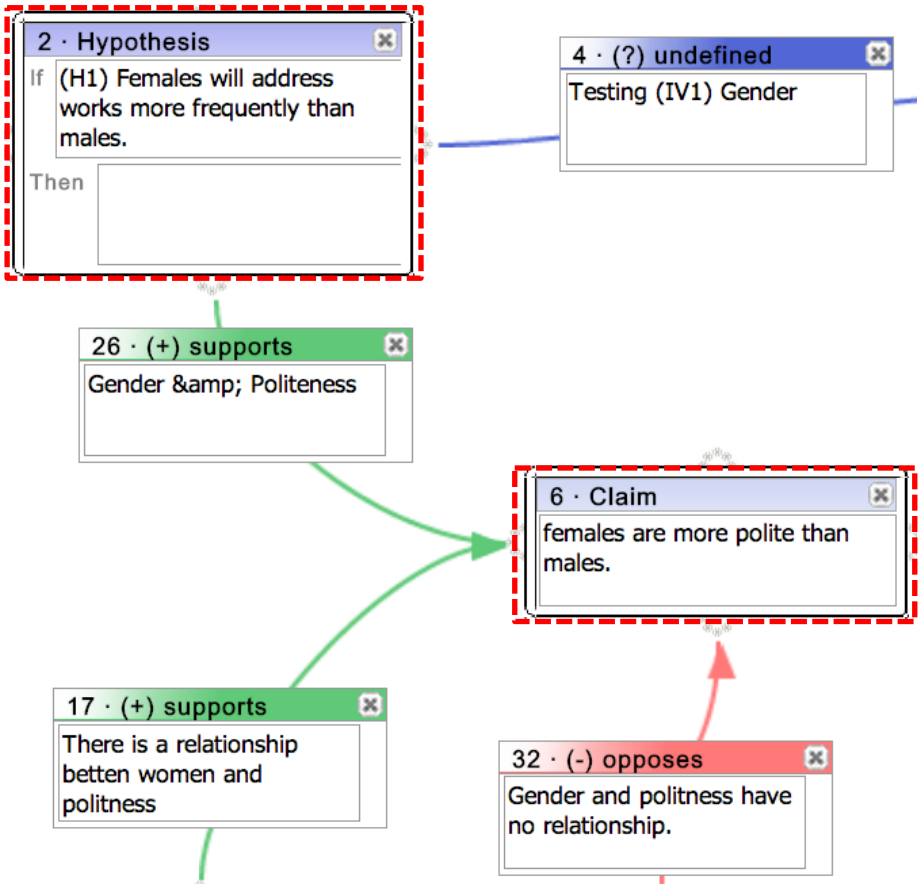


Location Patterns Algorithm

Location Patterns

- **Textual component content**: text in diagram node/arc are made concise

- Reviewers use **textual content in conjunction with node/arc type**
  - *gender hypothesis*
  - *claim that women are more polite than men*

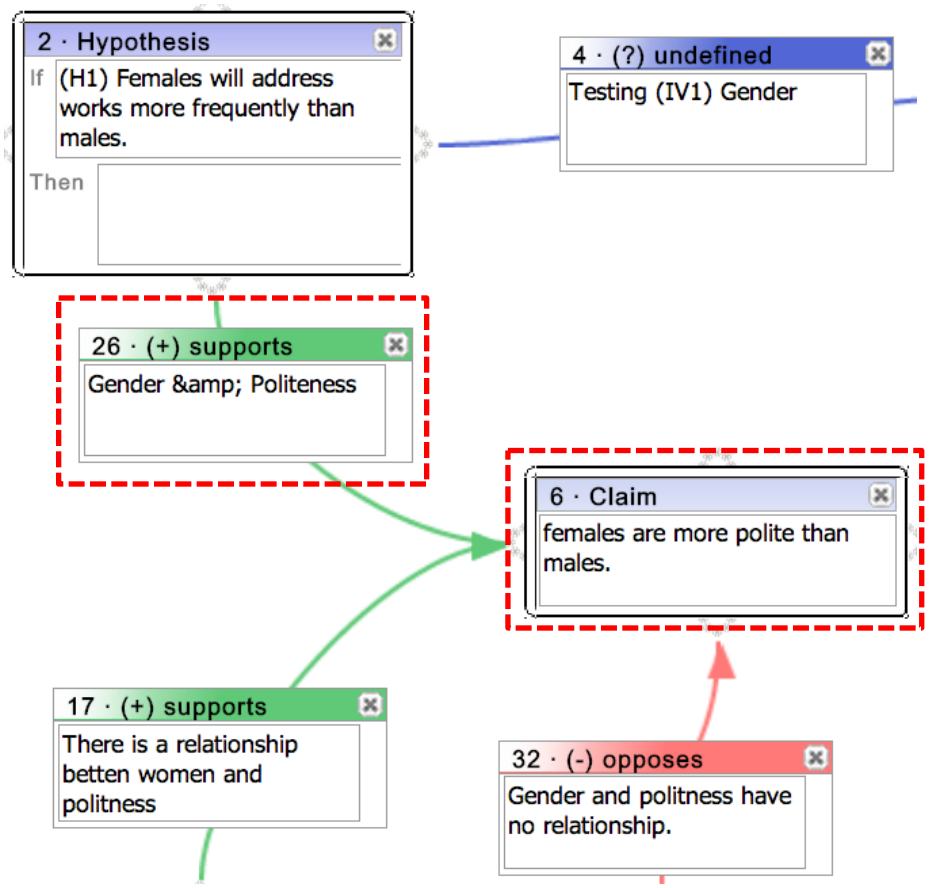


Location Patterns Algorithm

Location Patterns

- **Connected component**: referring to a line of argumentation

- Identify **connection between components**
  - *support for the gender hypothesis*
  - *claim node in between the opposes and support arcs 26 and 32*

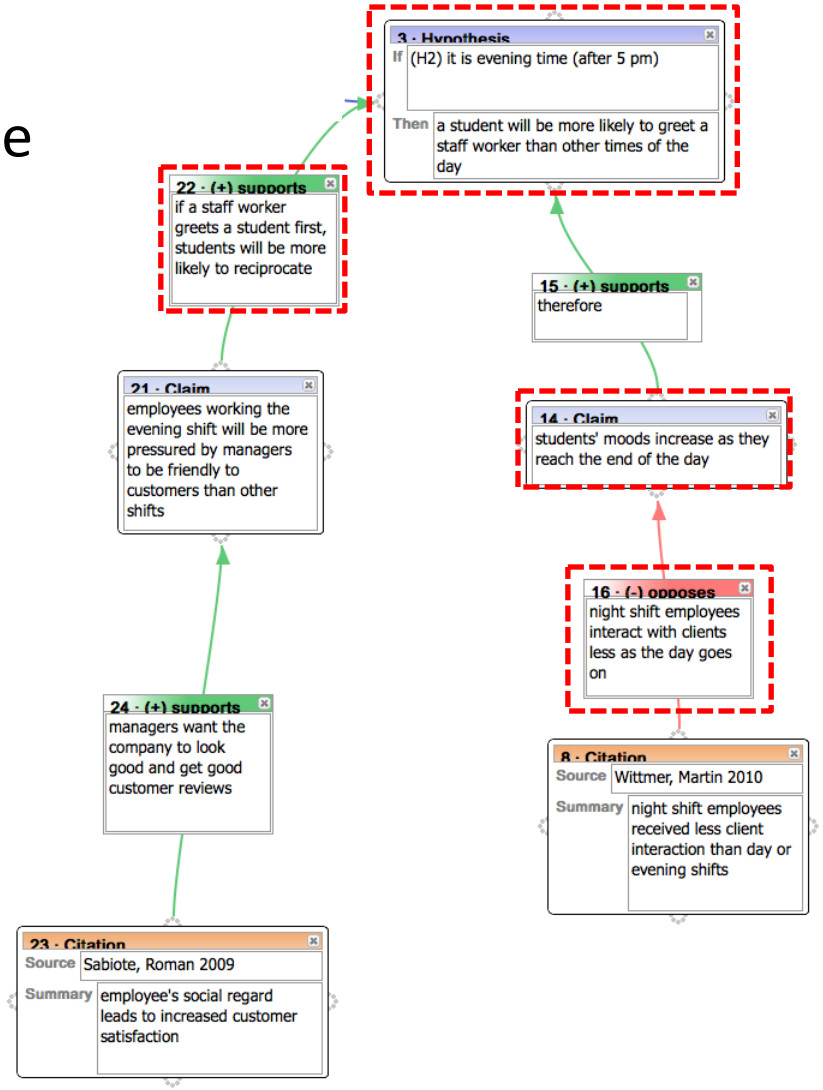


Location Patterns Algorithm

Location Patterns

- **Unique Component**: identifying the unique node/arc of a given type
  - *The opposing arc*

- **Typical numerical expressions** are used to express localization
  - *The second hypothesis, H2*
  - *[14] (claim node), #22 (support arc)*



Location Patterns Algorithm

Localization Pattern Algorithm (LPA)

- Location information must involve diagram component **keyword** surrounded by **supporting words**

- A diagram component keyword:
  - The words *node* or *arc*
  - Node/arc type from the ontology (*parsed automatically*)
- Supporting words are in proximity of a keyword which help locate the component

Localization Pattern Algorithm

- Supporting words are selected from common words between review and node/arc content (*stemmed already*)
- Identified accordingly to 5 localization pattern (*applied to review sentences that have common words*)
- **Numbered ontology type**: supporting words are number/list of numbers right after keyword
- **Textual component content**:
  - Supporting words occur right before keyword
  - Or after keyword with distance less than 3

Localization pattern algorithm

- **Unique component**: count number of node/arc of each type while parsing argument diagrams
- **Connected component**: extend node/arc text by the textual content of the other node/arc that it connects to
  - Supporting words must be in the extended content
- **Typical numerical expressions**: use held-out development data to learn regular expressions

Features used in paper review localization

- Xiong and Litman 2011: studied syntactic features from the parsed dependency tree of sentence
- Domain word count (**dw_cnt**)
  - dictionary of domain word is learned automatically from set of argument diagrams
- **So_domain**: indicates whether domain word appear between subject and object of review
- **Det_count**: counts number of demonstrative determiners in comment
- Overlapping window features:
  - Compute the maximal overlapping window
  - Report window size (**wnd_size**) and number of common words (**overlap_num**)

Experimental results

- Two baseline models
  - **Majority** model (*simply assign every instance label of the most common class*)
  - **pLocalization** model using only paper review features
    - Syntactic features vs. structural patterns
- Two proposed models:
  - **LPA**: use only output of LPA to identify the labels
  - **Combined**: add LPA binary output as a feature into pLocalization
- Models are learned using decision tree (*Weka J48*)
- Evaluated via 10-fold cross validation

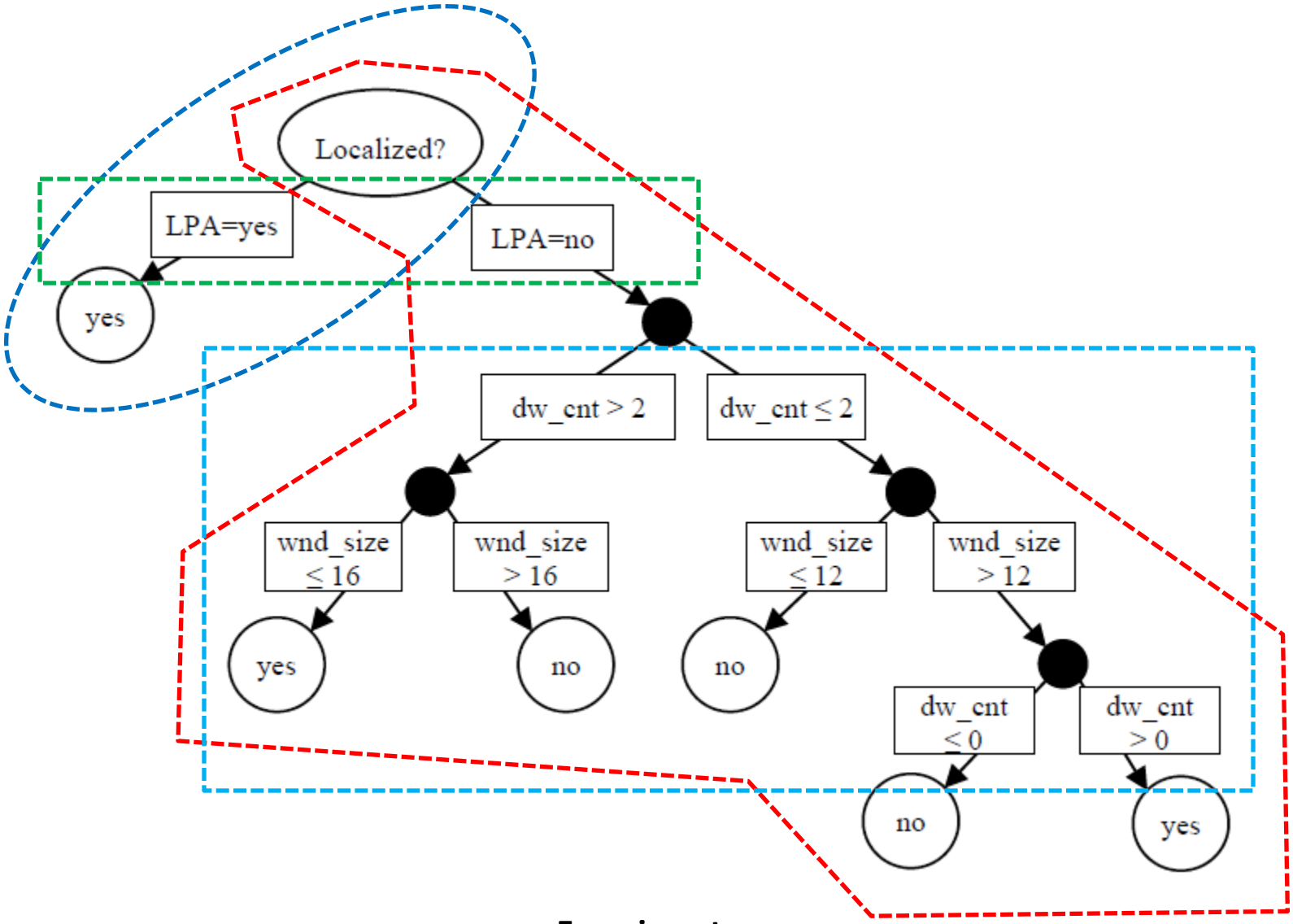
Experimental results

Metric	Majority	pLocalization	LPA	Combined
Accuracy (%)	74.07	73.98	80.34 *	83.78 *
Kappa	0	< 0.01	0.54 *	0.56 *
Weighted precision	0.55	0.55	0.83 *	0.84 *
Weighted recall	0.74	0.74	0.80 *	0.84 *

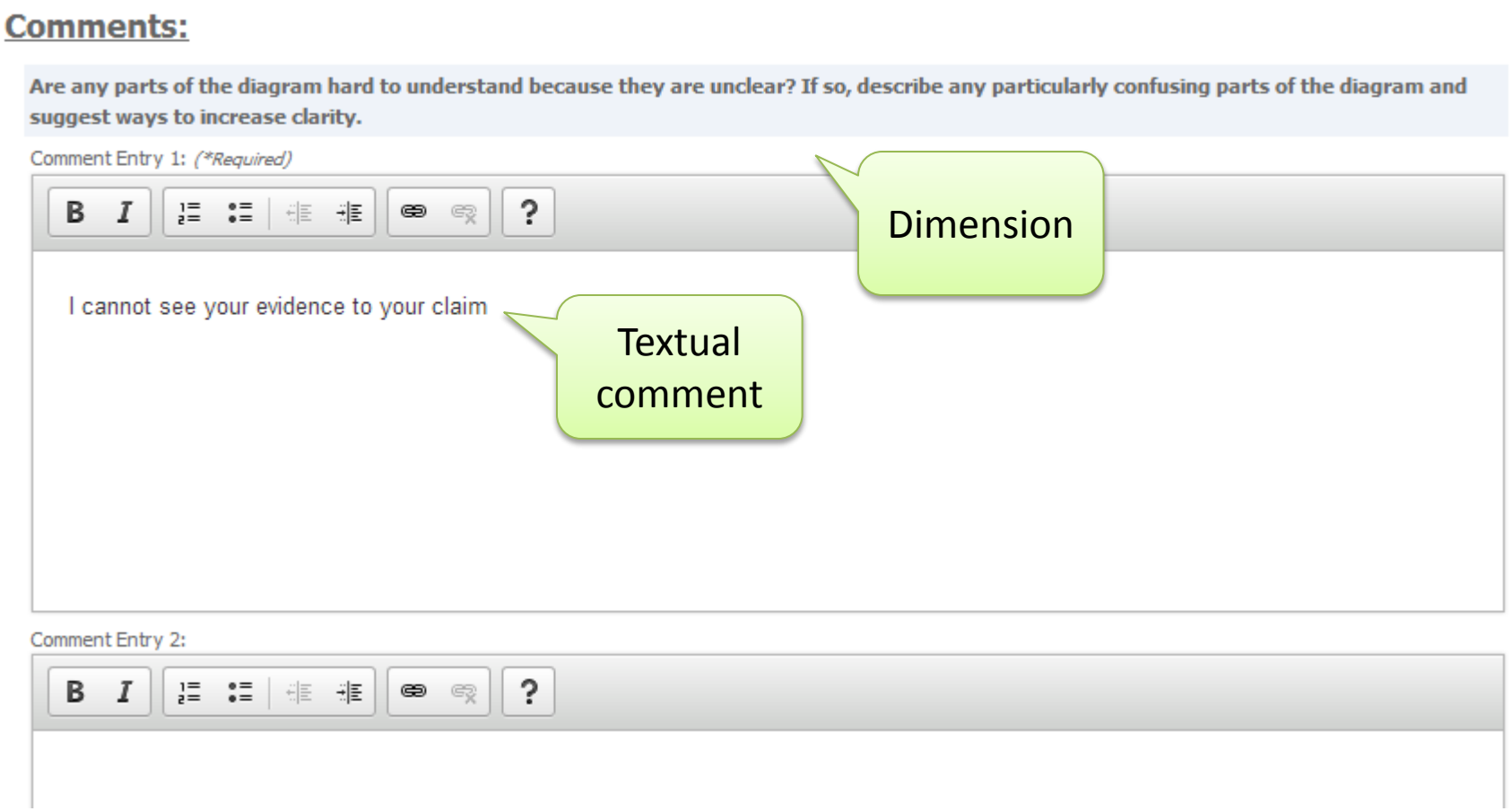
*: *significantly better than Majority*

- pLocalization does not outperform Majority
- LPA alone is significantly better than baselines
  - LPA can predict efficiently the minor class
- Combined model yields the best results of all

Learned decision tree



Integrating LPA into SWoRD



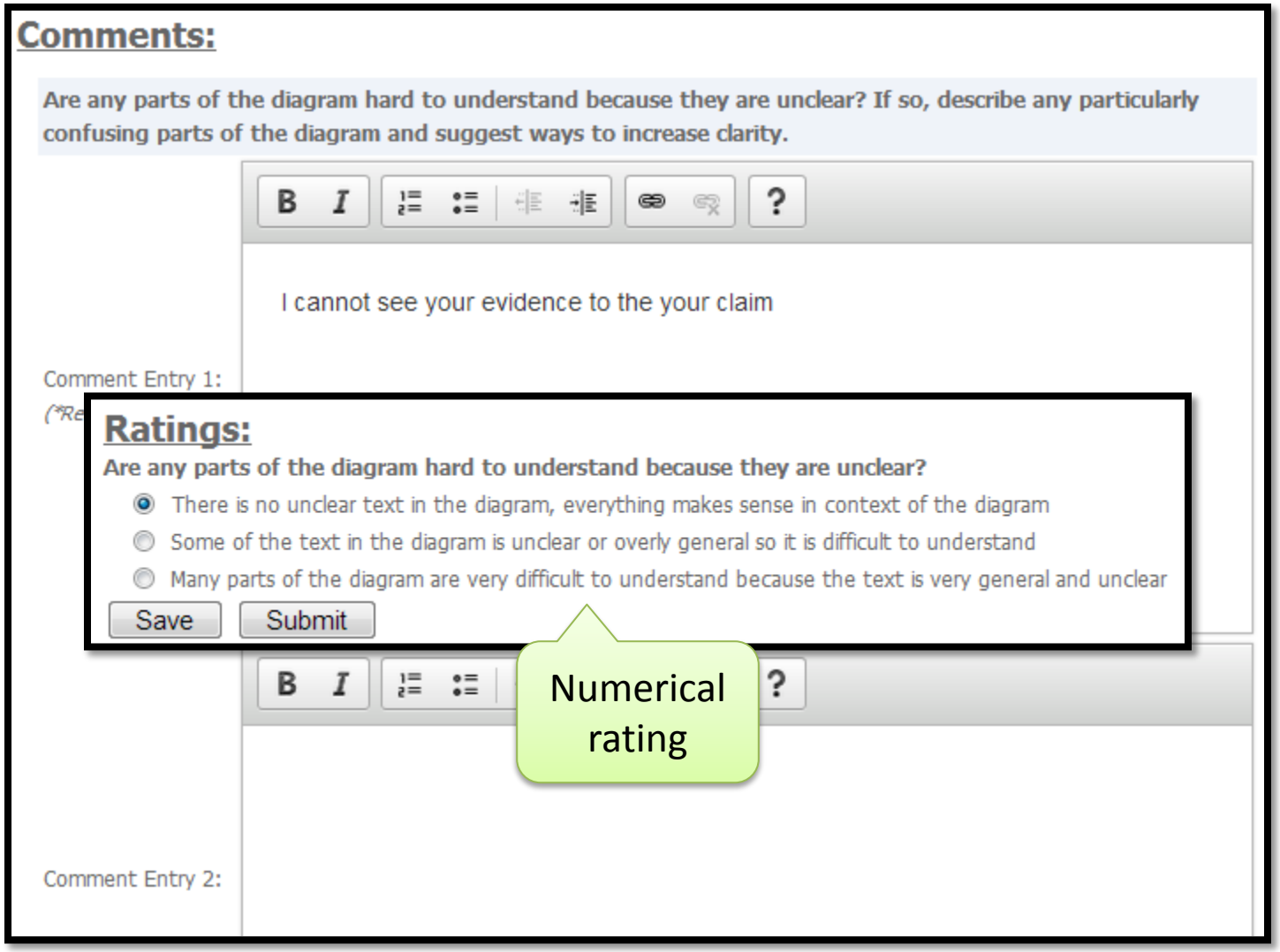
Screenshot of system intervention

Conclusion and future work

- LPA algorithm for identifying localization in peer review of argument diagrams
  - Outperforms a model developed for paper review localization
  - Combining the two approaches work best of all
- Deployed in SWoRD in June 2013
- In future, automatically learn patterns and regular expressions
- Test on new corpus with different ontology
- Apply lesson learned from developing LPA back to paper review localization model

THANK YOU

Questions
and
Comments


Comments:
Are any parts of the diagram hard to understand because they are unclear? If so, describe any particularly confusing parts of the diagram and suggest ways to increase clarity.
I cannot see your evidence to the your claim
Comment Entry 1:
Ratings:
Are any parts of the diagram hard to understand because they are unclear?
There is no unclear text in the diagram, everything makes sense in context of the diagram
Some of the text in the diagram is unclear or overly general so it is difficult to understand
Many parts of the diagram are very difficult to understand because the text is very general and unclear
Save
Submit
Numerical rating
Comment Entry 2:

Selected examples

Example #1	*(1) Specifying node(s) and/or arc(s) in the author's diagram to which your comment refers* The **citation [name w/wo year]** seems like a valuable study for the experiment. Merge the **node/node-type [node's ID]** with it's supporting arc from the citation. Your **conflicting/supporting node-type** is really solid!
Example #2	*(2) Quoting the excerpt from the author's textual content of node and/or arc to which your comment refers* For the other hypothesis I don't see evidence that supports **the statement that energy decreases in the afterno** **The claim that states Women are more inclined to express pro-social behaviors in a social setting** is A claim should be made that supports **the hypothesis that states women will be more polite compared t** For your **citation that talks about body chemistry and cortisol levels,** you should clarify how that is relate
Example #3	*(3) Referring explicitly to the specific line of argumentation that your comment addresses* I can see how **citation [citation's ID] is a good oppositional source to the first hypothesis.** Why does **claim [claim's ID] support the idea that people will be more polite in the evening?** I don't see **evidence that supports the statement that energy decreases in the afternoon.** To add **opposing arguments to the second citation** I suggest searching for something that says mood is wo I feel like the **[name w/wo year] citation can almost be a supporting citation for hypothesis [hypothesis's ID]** rather than just a comparison.

Type + ID/function

Type + content

Connection path