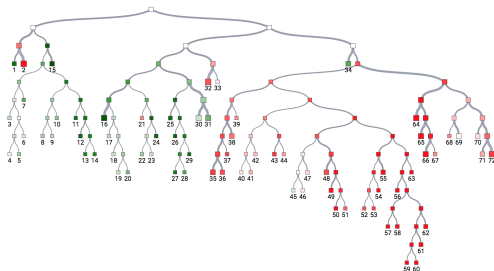


MEGA RST Discourse Treebanks with Structure and Nuclearity from Scalable Distant Sentiment Supervision

Patrick Huber and Giuseppe Carenini



University of British Columbia

{huberpat, carenini}@cs.ubc.ca

Linguistic View: Textual Data is Tree-Structured

- Within a document: Discourse-trees
 - Reveal structure underlying coherent documents
 - Groupings/Relations between clauses/sentences/paragraphs
 - Structure postulated by discourse theory:
 - **Rhetorical Structure Theory (RST)**[MT88]
 - PDTB [PDL⁺08]

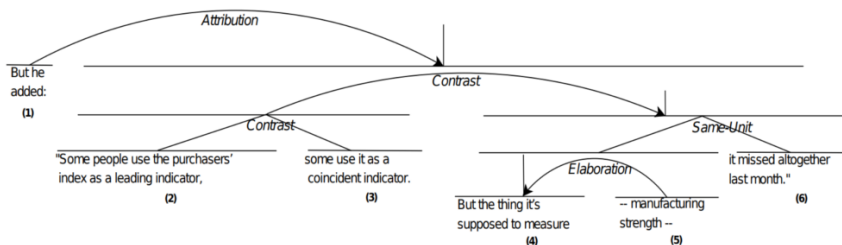


Figure: Complete, hierarchical RST-style discourse trees

¹<https://ntunlp.sg.github.io/project/parser/parser/>

Questions on RST and Discourse Parsing

1. What are “silver-standard” and “gold-standard” treebanks?
2. **What is nuclearity and why is it important in RST discourse parsing?**
3. The paper states that a nuclearity attribute encodes the importance of the node in its local context. What does the “importance of a node in its local context” mean? How is this determined?
4. **How are EDU’s segmented? How are those boundaries decided? It seems like the choice of boundaries would influence the subsequent calculations of sentiment polarity.**
5. What do the different nuclearity classes mean? Why can we have N-S, S-N, and N-N but not S-S?
6. **Why is training data for discourse parsing scarce?**
7. **I’m still a bit confused on how human-annotated discourse corpora are created? Is it something that a human has to do completely by hand?**
8. It is mentioned that the solution could be extended to predict discourse relations beside structure and nuclearity – which relationships are worth the additional analysis? What are some examples of discourse relations?
9. Are there other proposed or existing solutions in addition to the one proposed in this paper to address the scarcity of human annotated discourse treebanks as training data?
10. **Why did you decide to use trees (binary trees it seems by the images) instead of cyclic algorithms?**
11. What algorithms did previous top-performing discourse parsers employ?

What can Discourse be used for?

- **Sentiment analysis** [BJE15, NCN17]
- Summarization [GMC⁺14]
- Text classification [JS17]

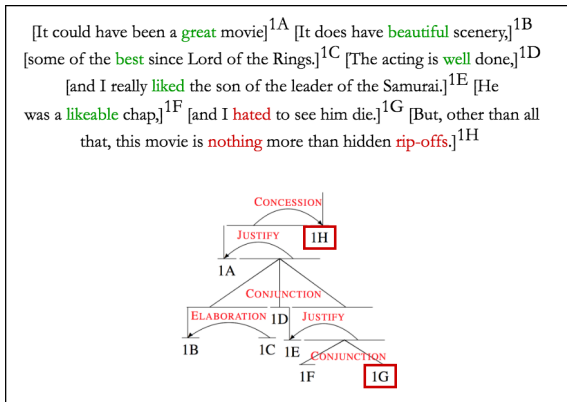
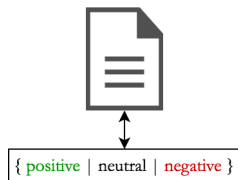
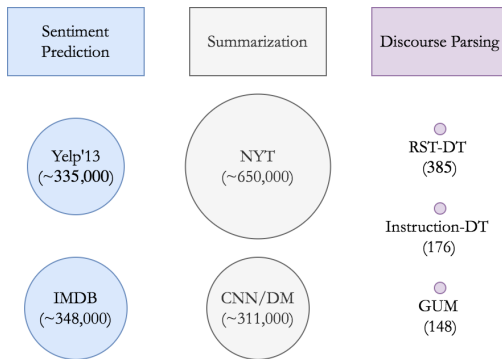


Figure: Adapted from [VT07]

The Problem with Annotated Data

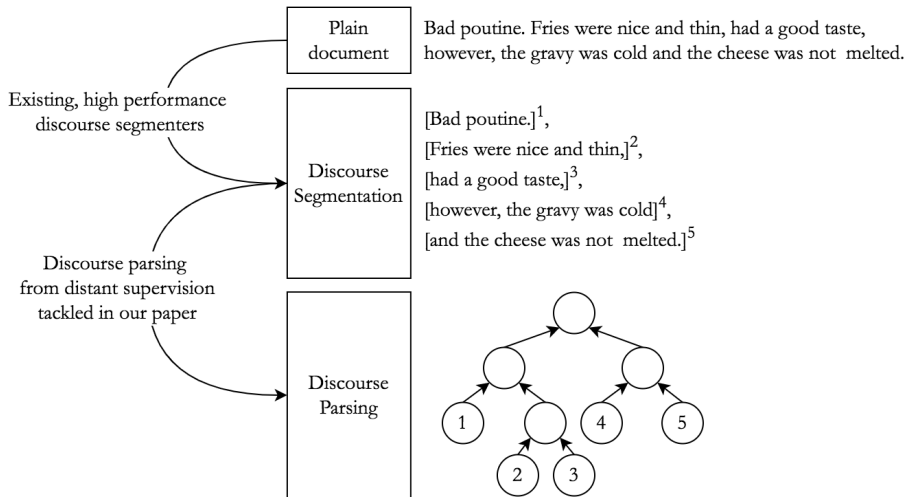
- Lack of annotated datasets (expensive and time consuming)
 - Corpora small and genre specific (News, Instructions...)
- ⇒ Limited applicability of deep learning methodologies



Questions on Datasets / Treebanks

1. **Why are the document numbers in Table 2 so varying between the treebanks?**
2. Would the efficacy of this method change depending on the type of data it was looking at? I.e. this was trained on yelp data, would it have better results if the subject matter was more specific like a medical database?
3. What was the reasoning for choosing to test against the human-annotated treebanks with a 90-10 training to testing split?
4. **Is there a way to crowdsource discourse annotations to increase the amount of training data and help advance the research?**
5. Why was the Yelp 2013 corpus used for creating MEGA-DT? What are the effects of using other corpuses, and was this explored at all?
6. To what extent would increasing the size of the treebank improve the performance of the parser, even if the quality of converting from sentiment to discourse of the corpus is not very high?

The Sub-Tasks of Discourse Parsing

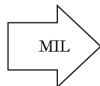


From Plain EDUs to Discourse Structures (1/2)

[Bad poutine.]¹,
[Fries were nice and thin,]²,
[had a good taste,]³,
[however, the gravy was cold]⁴,
[and the cheese was not melted.]⁵

+

★☆☆☆☆ (Negative)



[Bad poutine.]¹,
[Fries were nice and thin,]²,
[had a good taste,]³,
[however, the gravy was cold]⁴,
[and the cheese was not melted.]⁵

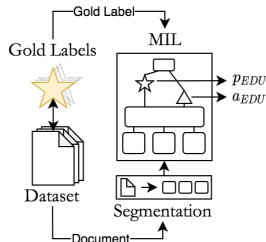
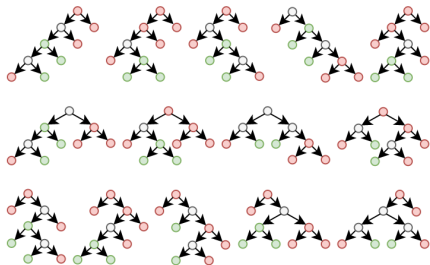


Figure: Multiple Instance Learning (MIL) as presented in [AL18], used in [HC19]

From Plain EDUs to Discourse Structures (2/2)

[Bad poutine.]¹,
 [Fries were nice and thin.]²,
 [had a good taste.]³,
 [however, the gravy was cold]⁴,
 [and the cheese was not melted.]⁵



$$p_{i,j} = \frac{p_i * a_i + p_j * a_j}{a_i + a_j} \quad a_{i,j} = \frac{a_i + a_j}{2}$$

$$t^* = \operatorname{argmin}_{t \in T} \Delta(p_t, p_{\text{gold}})$$

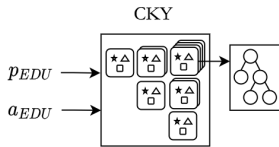


Figure: CKY as presented at EMNLP 2019 [HC19]

Questions on the General Algorithm

1. **In each cell of the matrix of EDUs, what information do the subtrees contain?**
2. Why are attention values used to capture nodes' relative importance in the tree, and how are they measured?
3. How does adding an additional subtree at every merge in the CKY procedure represent N-N nuclearity? How do we use polarity scores to classify N-N?
4. **How do we get the attention values of the leaf notes? A table that has the attention value for each word (terminal terms)?**
5. For predicting short discourses, are there more efficient strategies?
6. Can we use a machine learning approach to determine how many subtrees to preserve in each cell (B)? (maybe by running a very large number of experiments with different B values)
7. **How is the CKY algorithm from class different from what is used in the paper?**
8. The best performing approach for aggregating its two child nodes to find sentiment polarity p using the attention score a . A nodes attention score can be calculated independent of the same node's sentiment polarity. How is the sentiment polarity p related to the attention score a ?
9. How complex it is to implement the proposed strategy compared to the CKY algorithm? not sure if there is a trade-off between CKY algorithm and the proposed strategy

Is the Standard CKY Approach Sufficient?

Advantages:

Time complexity $O(n^3)$

Global Optimum

Drawbacks:

Space complexity $C_n = \frac{1}{n+1} \binom{2n}{n}$

\forall Sub-trees \rightarrow Gold-labels

- Full CKY approach $\rightarrow \leq 20$ EDUs²
- Generated treebank only $\approx 100,000$ (out of 250,000) documents
- No additional attributes (e.g., Nuclearity, Relations)

²Using our modern servers

Questions on Time/Space Complexity

1. Elaborate on the time complexity of the original solution to this problem and how using beam search reduces it?
2. **How does the time complexity of this model compare with that of other top-performing models?**
3. **Why is there no analysis of the computational complexity of the new algorithm vs. the old one? Is computational complexity not as important in this context as spatial complexity?**
4. What is the time complexity for this algorithm? How long did the experimental datasets and training take?
5. If the space complexity using beam search is so low, would it be feasible to consider another algorithm which focuses on time complexity or do we not care about time complexity as much as we care about space complexity?

Our new Approach (1/3)

Beam-search as effective heuristic approach for tree-structured problems:

- Syntactic parsing [VKK⁺15, FSK17, DKBS16]
- Discourse parsing [MRCV19]

⇒ Apply beam-search to CKY, limiting #sub-trees per cell to B

Beam	20 EDUs	30 EDUs	100 EDUs
1	1.6KB	3.7KB	40KB
10	24KB	48KB	440KB
100	920KB	1.5MB	7.9MB
∞	3.6GB	1.9PB	400SB

Table: KB = 10^3 , MB = 10^6 , GB = 10^9 , PB = 10^{15} , SB = 10^{54}

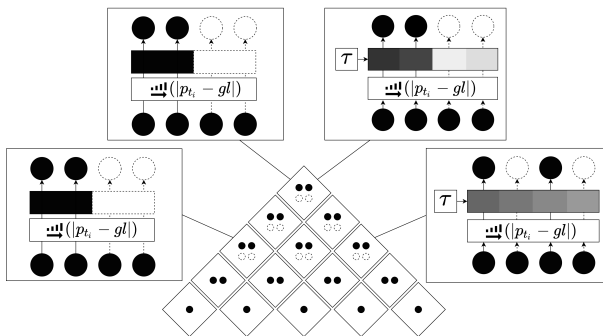
Questions on the Heuristics/Beam-Search

1. **Are there any non-heuristic approaches that tackle the same issue? If so, how successful are these approaches?**
2. Do you see this being expanded with other search strategies other than beam search as well? / Is there any other heuristic rules that could perform similar or better than the one used in this paper?
3. If we have “infinite resources” i.e. “infinite memory” and “infinite time” would it be possible to come up with another method that performs better than the one in the paper?
4. **Other than beam size, what other heuristics were considered - why not use one based on depth or a certain approximating formula?**
5. **Assuming that the distance between gold-label sentiment and model prediction decreases as B increases, why was $B=10$ chosen over 15 or 20?**
6. Why didn't you try other heuristics to test how well the proposed heuristic works?
7. How does changing the rule of picking B subtrees influence the result?
8. Does this method have a maximum limitation? (for space or time consumption)? What if a longer documents need to be processed?
9. Beam search is not necessarily optimal (that is, there is no guarantee that it will find the best solution). In general, beam search returns the first solution found. Could this have led to incorrect subtrees being generated by MEGA-DT?

Our new Approach (2/3)

Stochastic Component:

- Why? – Strict enforcement of gold-level sentiment not preferable on low sub-trees
- How? – Softmax selection using Boltzmann–Gibbs distribution & Depth-dependent stochastic trade-off τ



CKY Matrix

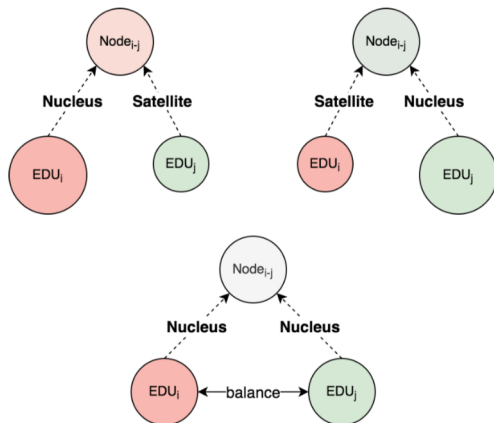
Questions on the Stochastic Approach/Explore-Exploit

1. Can the algorithm be improved via tweaking some of the parameters of the exploitation-exploration tradeoff calculations?
2. What technique or improvement could be made in order to promote high degrees of exploration on the low levels of the subtrees, whilst simultaneously heavily enforcing that the sentiment of the subtrees align with the overall document sentiment, and how would this improve the performance?
3. How would we be able to calculate the trade off between exploration and exploitation for their proposed solution?
4. **How was tau selected? Could there potentially be more efficient functions? Why a linear function?**
5. What would happen if the lower levels of the binary trees have a high degree of exploitation instead of exploration and vice versa?
6. **Given the stochasticity, is it guaranteed that Stochastic Beam Search will find a global optimum?**
7. **Why was soft-max selection using the Boltzmann-Gibbs distribution chosen over an epsilon greedy approach or other exploration-exploitation methods?**
8. Is there any way for the algorithm to make sure that it doesn't discard the best subset during exploration?
9. Are there alternatives to the technique used to simulate exploration-exploitation, which could more intelligently consider the sentiment of smaller portions of the document when choosing subtrees to keep?

Our new Approach (3/3)

Add nuclearity as additional feature:

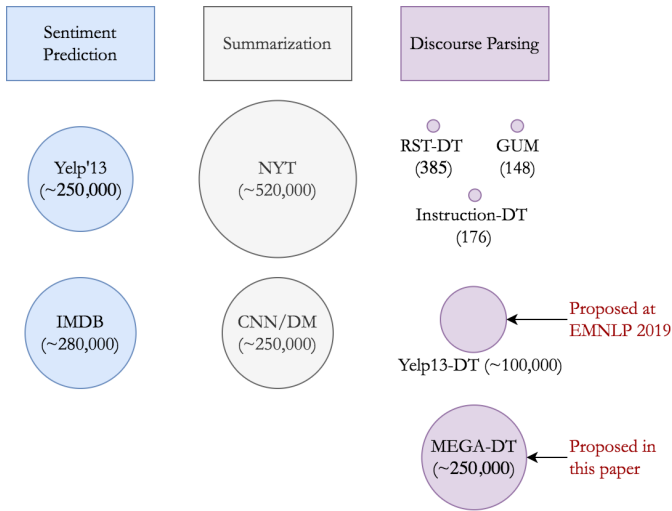
- Shown to be important for downstream tasks [Mar00, JS17, SQ19]
- Moving towards complete discourse trees



Questions on Nuclearity Improvements

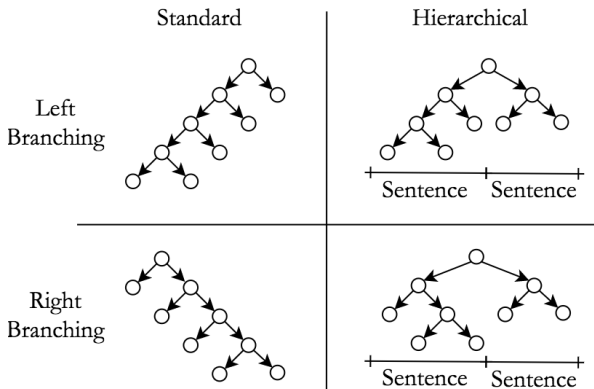
1. **What is causing the large amount of over-prediction of the N-N nuclearity class? How would we be able to improve on this?**
2. Given the definition of nuclearity provided in the work, that is, a measure of “importance” how many methods are there of deriving it given some input?
3. **What is the reason that only considering two out of three nuclearity classes(N-S and S-N) generates a poor performance on the nuclearity classification?**
4. Is there a way to help us distinguish between nucleus and satellite sentences?
5. **Taking the average of the attention values of a parent’s children may be too simple ? Is there another way to calculate the attention value of a parent? For example, choosing the weight the attention value of one child more than another child?**

Our new Approach Result



Evaluation - Simple Baselines (1/2)

Four simple baselines for structure prediction:



Simple baseline for nuclearity prediction:
Majority class (from training dataset)

Evaluation - Simple Baselines (2/2)

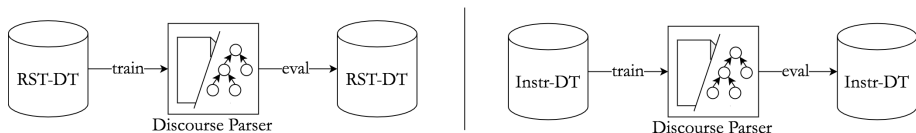
Approach	Structure		Nuclearity	
	RST-DT	Instr-DT	RST-DT	Instr-DT
Right Branching	54.64	62.72	×	×
Left Branching	53.73	52.16	×	×
Hier. Right Branching	74.37	75.34	×	×
Hier. Left Branching	70.58	63.75	×	×
Majority Class	×	×	(N)61.33	(N) 76.48

Results of the micro-averaged precision measure on RST-parseval.
Best performance per sub-table is **bold**, × not feasible.

Evaluation - Intra-Domain Systems (1/2)

Intra-domain (train/test on **same** domain/treebank) is standard evaluation
→ Large variety of previous work, our baselines include:

DPLP[JE14], gCRF[FH14], CODRA[JCN15], Li[LLC16],
Two-Stage[WLW17], Yu[YZF18]



Evaluation - Intra-Domain Systems (2/2)

Approach	Structure		Nuclearity	
	RST-DT	Instr-DT	RST-DT	Instr-DT
Right Branching	54.64	62.72	×	×
Left Branching	53.73	52.16	×	×
Hier. Right Branching	74.37	75.34	×	×
Hier. Left Branching	70.58	63.75	×	×
Majority Class	×	×	(N)61.33	(N) 76.48
Intra-Domain Evaluation				
DPLP[JE14]*	82.00	–	68.20	–
gCRF[FH14]*	84.30	–	69.40	–
CODRA[JCN15]*	82.60	82.88	68.30	64.13
Li[LLC16]*	82.20	–	66.50	–
Two-Stage[WLW17]	86.00	79.43	72.40	62.39
Yu[YZF18]	85.50	–	73.10	–

Results of the micro-averaged precision measure on RST-parseval.

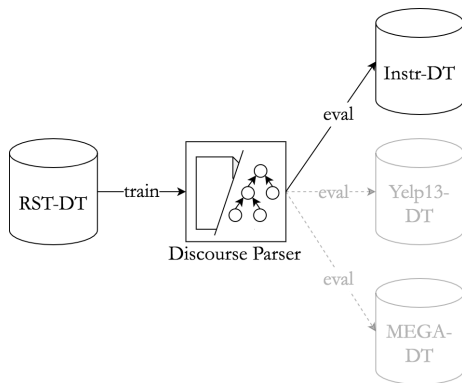
Best performance per sub-table is **bold**, * results taken from [MMA17],

– not published, × not feasible.

Evaluation - Inter-Domain Systems (1/5)

Inter-domain (train/test on **different** domains) is new, more difficult and more insightful

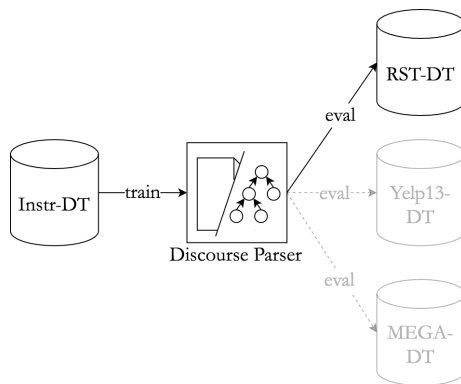
Experiments using the Two-Stage parser [WLW17]



Evaluation - Inter-Domain Systems (2/5)

Inter-domain (train/test on **different** domains) is new, more difficult and more insightful

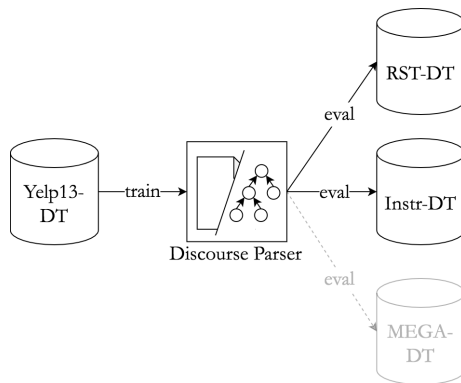
Experiments using the Two-Stage parser [WLW17]



Evaluation - Inter-Domain Systems (3/5)

Inter-domain (train/test on **different** domains) is new, more difficult and more insightful

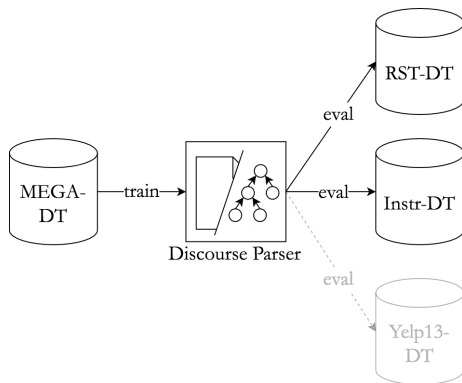
Experiments using the Two-Stage parser [WLW17]



Evaluation - Inter-Domain Systems (4/5)

Inter-domain (train/test on **different** domains) is new, more difficult and more insightful

Experiments using the Two-Stage parser [WLW17]



Questions on Intra-/Inter-Domain

1. What is the difference between inter-domain and intra-domain? Why are there be statistically significant improvements on inter-domain discourse prediction?
2. **Could you elaborate more about what it means for the model to do well with Intra-Domain evaluation vs Inter-Domain.**
3. **What contributes to the difference in performance of training and testing for inter vs intra domain discourse prediction?**
4. Is the weaker intra-domain performance a consequence of the trade-offs made in terms of inter-domain performance?
5. How come the parser did the best on inter-domain experiments as opposed to intra-domain experiments?
6. Why is is that, if intra-domain tasks are easier, a parser trained on MEGA-DT does not match the performance of training and testing on the same treebank? If it performs well on inter-domain tasks, why does performance decline for intra-domain tasks?

Evaluation - Inter-Domain Systems (5/5)

Approach	Structure		Nuclearity	
	RST-DT	Instr-DT	RST-DT	Instr-DT
Right Branching	54.64	62.72	×	×
Left Branching	53.73	52.16	×	×
Hier. Right Branching	74.37	75.34	×	×
Hier. Left Branching	70.58	63.75	×	×
Majority Class	×	×	(N) 61.33	(N) 76.48
Intra-Domain Evaluation				
DPLP[JE14]*	82.00	–	68.20	–
gCRF[FH14]*	84.30	–	69.40	–
CODRA[JCN15]*	82.60	82.88	68.30	64.13
Li[LLC16]*	82.20	–	66.50	–
Two-Stage[WLW17]	86.00	79.43	72.40	62.39
Yu[YZF18]	85.50	–	73.10	–
Inter-Domain Evaluation				
Two-Stage _{RST-DT}	×	73.57	×	49.78
Two-Stage _{Instr-DT}	74.32	×	44.68	×
Two-Stage _{Yelp13-DT[HC19]}	76.41	74.14	35.72	33.35
Two-Stage _{MEGA-DT}	† 77.82	† 75.18	44.88	† 54.87
Human [MMA17]	88.30	–	77.30	–

Results of the micro-averaged precision measure on RST-parseval.

Best performance per sub-table is **bold**, * results taken from [MMA17],

† statistically significant (Bonferroni adjusted), – not published, × not feasible.

Evaluation - Ablation / Dataset-Sizes

Approach	Structure		Nuclearity	
	RST-DT	Instr-DT	RST-DT	Instr-DT
Two-Stage _{MEGA-DT-Base}	75.92	73.87	36.46	34.48
Two-Stage _{MEGA-DT +Stoch}	77.58	73.76	37.43	35.89
Two-Stage _{MEGA-DT +Nuc}	76.76	74.35	44.22	54.10
Two-Stage _{MEGA-DT}	77.82	75.18	44.88	54.87

Table: Ablation study, measured as the micro-average precision on RST-parseval.

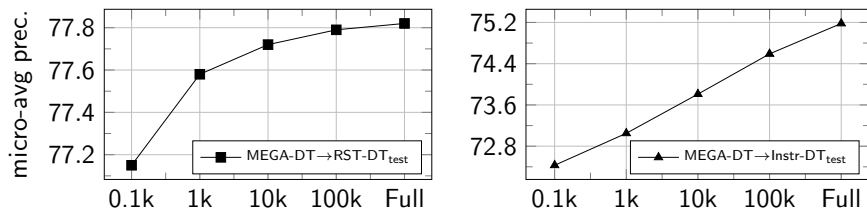
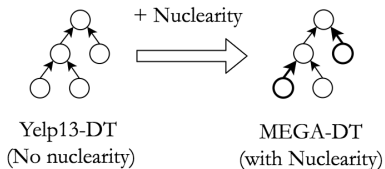
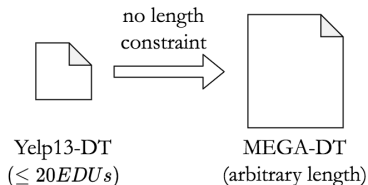
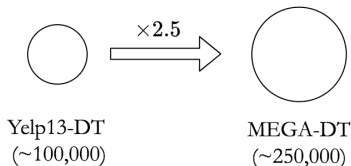


Figure: Performance-trend over increasingly large subsets.

Questions on Experiments

1. **Table 4 is referred to as an 'ablation study'. What does that mean? Does that mean 'how do the results change as we remove the components of this algorithm'?**
2. If we were to remove or weaken the strategy used for exploration and exploitation trade-off, will parsers trained on MEGA-DT still be better than other treebanks that are available?
3. **Why did they choose to compare MEGA-DT to the other treebanks mentioned in the research paper and not a more exhaustive list of treebanks?**
4. Even though Two-Stage was the best parser in intra-domain evaluation for the most part, why not also train the other parsers on Mega-DT and compare their inter-domain performances as well?
5. **In the evaluation, why do they use a randomly selected subset containing 10,000 documents from the Yelp's 13 dataset?**
6. Why do you think training a parser on MEGA-DT could not match the performance on the intra-domain?
7. Can similar techniques be used for other AI training data outside of discourse parsing?

Conclusion (1/2)



Possible due to:



Beam-search



Stochastic component
(Exploration vs. Exploitation)

Conclusion (2/2)

- Enable augmentation of any sentiment-annotated dataset
- Allows generation of large-scale domain/genre-specific discourse treebanks.
- MEGA-DT is our case study for the effectiveness
- Published dataset & code! Check out:
<https://www.cs.ubc.ca/cs-research/lci/research-groups/natural-language-processing/>

Questions on Scalability

1. **This method is quite scalable, how would a much larger treebank (e.g. 2,500,000 documents instead of 250,000) perform?**
2. How much further could this method be scaled? I.e. could significantly larger corpora than MEGA-DT be generated?
3. MEGA-DT is based on Yelp '13, isn't there a possibility that a newer dataset would help improve the performance of this work?

Questions on Auxiliary Tasks

1. **One of the extensions for the work mentions that a long term goal is to explore auxiliary tasks for distant supervision of discourse, like summarization, question answering and machine translation. How different would the approach to solving these problems be?**
2. Could this work apply to other supervision methods other than overall sentiment? How widely applicable is this in regard to whether most real-life datasets already have sentiment-annotation or if this method only applies to a smaller subset of databases.

Evaluation on Downstream Tasks - Setup

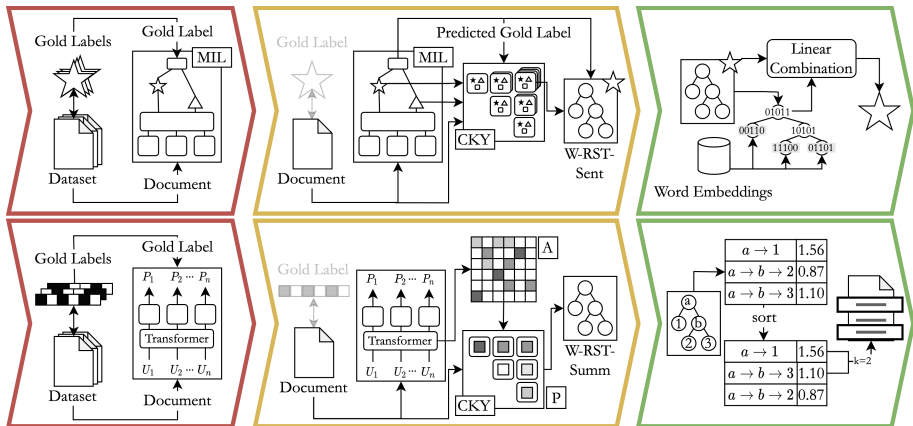


Figure: Three phases of the approach. Left/Center: Detailed view into the generation of weighted RST-style discourse trees. Right: Downstream evaluation

Questions on Downstream Tasks

1. **How is nuclearity-attribution critical in informing downstream tasks, explicitly?**
2. One of the potential uses of MEGA-DT is to use it for sentiment analysis for long documents. Will there be a performance overhead as the document size increases?

Questions on the Sentiment Analysis Task

1. Why are only sentiment-based datasets used and no other types for evaluation?
2. **Why stop at 1 case study corpus, why not create a handful of MEGA-DT-like corpora in different genres to evaluate the approach?**
3. **If it is possible for the sentiment of a paragraph to be subjective, how are the resulting treebanks different for the different possible sentiment annotations?**
4. Is it possible for it to work with ongoing texts such as the news or conversation that happened right now. Would the accuracy be affected a lot?
5. How would MEGA_DT respond in the training of a discourse parser for non-sentiment datasets, where document-level non-sentiment information is present?





Questions on Different Languages

1. Could this RST-style discourse parsing be applied to other spoken/written languages besides english?
2. Would MEGA-DT creation techniques need to be created for each language / context?

Questions on Applications




1. How does this sort of algorithm help us do things; I see that it's about breaking a statement into EDU's and figuring out their structure based on sentiment of the whole and the parts, but what are the applications?
2. **Has the new MEGA-DT discourse treebank been used for anything yet?**
3. Can MEGA-DT substitute RST-DT in real life? If not why?
4. Will automatically generated discourse trees eventually produce better results than human-annotated ones?
5. Have other domain-specific treebanks been generated with this new approach and how did they compare to existing banks?
6. **Could the parser be used in other industries besides restaurants as readily?**
7. Being able to translate a model from one domain to a different domain is very important. Especially given that NLP techniques (and machine learning techniques in general) rely on data. Is that why you chose to evaluate MEGA-DT on inter-domains? I would imagine that it would make more sense to evaluate it only intra-domain since the data it's learning on is similar to what it will be tested on.

References I





-  Stefanos Angelidis and Mirella Lapata, *Multiple instance learning networks for fine-grained sentiment analysis*, Transactions of the Association for Computational Linguistics **6** (2018), 17–31.
-  Parminder Bhatia, Yangfeng Ji, and Jacob Eisenstein, *Better document-level sentiment analysis from rst discourse parsing*, Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 2212–2218.
-  Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A Smith, *Recurrent neural network grammars*, arXiv preprint arXiv:1602.07776 (2016).
-  Vanessa Wei Feng and Graeme Hirst, *A linear-time bottom-up discourse parser with constraints and post-editing*, Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, 2014, pp. 511–521.

References II

-  Daniel Fried, Mitchell Stern, and Dan Klein, *Improving neural parsing by disentangling model combination and reranking effects*, arXiv preprint arXiv:1707.03058 (2017).
-  Shima Gerani, Yashar Mehdad, Giuseppe Carenini, Raymond T Ng, and Bitia Nejat, *Abstractive summarization of product reviews using discourse structure*, Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1602–1613.
-  Patrick Huber and Giuseppe Carenini, *Predicting discourse structure using distant supervision from sentiment*, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 2306–2316.




-  Shafiq Joty, Giuseppe Carenini, and Raymond T Ng, *Codra: A novel discriminative framework for rhetorical analysis*, *Computational Linguistics* **41** (2015), no. 3.
-  Yangfeng Ji and Jacob Eisenstein, *Representation learning for text-level discourse parsing*, *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2014, pp. 13–24.
-  Yangfeng Ji and Noah A Smith, *Neural discourse structure for text categorization*, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 996–1005.

References IV

-  Qi Li, Tianshi Li, and Baobao Chang, *Discourse parsing with attention-based hierarchical neural networks*, Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 362–371.
-  Daniel Marcu, *The theory and practice of discourse parsing and summarization*, MIT press, 2000.
-  Mathieu Morey, Philippe Muller, and Nicholas Asher, *How much progress have we made on rst discourse parsing? a replication study of recent results on the rst-dt*, Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 1319–1324.
-  Amandla Mabona, Laura Rimell, Stephen Clark, and Andreas Vlachos, *Neural generative rhetorical structure parsing*, arXiv preprint arXiv:1909.11049 (2019).

-  William C Mann and Sandra A Thompson, *Rhetorical structure theory: Toward a functional theory of text organization*, Text-Interdisciplinary Journal for the Study of Discourse **8** (1988), no. 3, 243–281.
-  Bitá Nejat, Giuseppe Carenini, and Raymond Ng, *Exploring joint neural model for sentence level discourse parsing and sentiment analysis*, Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, 2017, pp. 289–298.
-  Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber, *The penn discourse treebank 2.0.*, LREC (2008).
-  Vighnesh Shiv and Chris Quirk, *Novel positional encodings to enable tree-based transformers*, Advances in Neural Information Processing Systems, 2019, pp. 12058–12068.

References VI

-  Oriol Vinyals, Łukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton, *Grammar as a foreign language*, Advances in neural information processing systems, 2015, pp. 2773–2781.
-  Kimberly Voll and Maite Taboada, *Not all words are created equal: Extracting semantic orientation as a function of adjective relevance*, Australasian Joint Conference on Artificial Intelligence, Springer, 2007, pp. 337–346.
-  Yizhong Wang, Sujian Li, and Houfeng Wang, *A two-stage parsing method for text-level discourse analysis*, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2017, pp. 184–188.



Nan Yu, Meishan Zhang, and Guohong Fu, *Transition-based neural rst parsing with implicit syntax features*, Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 559–570.

Thank you

Questions?