Constructing Imaginary Discourse Trees Improves Answering Convergent Questions

Boris Galitsky[1], Dmitry Ilvovsky[2], Gerhard Wohlgenannt[3]

[1]Oracle Inc., Redwood Shores, USA
Boris.galitsky@oracle.com
[2]National Research University Higher School of Economics, Moscow, Russia
dilvovsky@hse.ru
[3]International Laboratory of Information Science and Semantic Technologies
ITMO University, St. Petersburg, Russia
gwohlg@corp.ifmo.ru

Abstract. We introduce a concept of an *imaginary discourse tree* to improve question-answering (Q/A) recall for complex, multi-sentence, convergent questions. Augmenting a discourse tree of an answer with tree fragments obtained from ontologies, we obtain a canonical discourse representation of this answer that is independent of a thought structure of a given author. This mechanism is critical for finding an answer that is not only relevant in terms of questions entities but also in terms of inter-relations between these entities in an answer and its style. We evaluate the Q/A system enabled with imaginary discourse trees and observe a substantial increase of accuracy answering complex questions such as Yahoo! Answers and www.2carpros.com.

Keywords: rhetoric structure, discourse tree, question answering

1 Introduction

In spite of the great success of search technologies, the problem of leveraging background knowledge is still on the agenda of search engineering, for both conventional and learning-based systems. Background knowledge ontologies are difficult and expensive to build, and knowledge graphs – based approaches usually have a limited expressiveness and coverage. In this study we explore how a discourse analysis (which is domain-independent) can substitute certain features of ontology-based search. There are few popular discourse theories describing how DT can be constructed from the text. In our work we used Rhetoric Structure Theory (RST, [5]).

Ontologies are in great demand for answering complex, multi-sentence questions with a precise answer in such domain as finance, legal, and health. In educational domain this type of questions is referred to as convergent: answers to these types of questions are usually within a very finite range of acceptable accuracy. These may be at several different levels of cognition including comprehension, application, analysis, or ones where the answerer makes inferences or conjectures based on material read,

presented or known. Answering convergent questions is an underexplored Q/A domain that can leverage discourse analysis [27].

Discourse trees (DT) became a standard for representing how thoughts are organized in text, in particular in a paragraph of text, such as an answer. Discourse-level analysis has been shown to assist in a number of NLP tasks where learning linguistic structures is essential [24, 29]. DTs outline the relationship in between entities being introduced by an author. Obviously, there are multiple ways the same entities and their attributes are introduced, and not all rhetoric relations that hold between these entities occur in a DT for a given paragraph.

When DTs are used to coordinate questions and answers, we would want to obtain an "ideal" DT for an answer, where all rhetoric relations between involved entities occur. To do that, we need to augment an actual (available) DT of answer instance with a certain rhetorical relations which are missing in the given answer instance but can be mined from text corpora or from the web. Hence to verify that an answer A is good for a given question Q, we first verify that their DTs (*DT-A* and *DT-Q*) agree and after that we usually need to augment the DT-A with fragments of other DTs to make sure all entities in Q are communicated (addressed) in augmented DT-A.

Hence instead of relying on an ontology that would have definitions of entities which are missing in a candidate answer we mine for the rhetorical relations between these entities online. This procedure allows us to avoid an offline building of bulky and costly ontologies. At the same time, the proposed approach can be implemented on top of a conventional search engine.

The paper structure is organized as follows. In Section 2 we mention some of the related works. In Section 3 we describe the concept of ***imaginary discourse tree*** and consider a number of examples illustrating how it can be used and constructed. In Section 4 we propose Q/A filtering algorithm which is the core part of our approach. In Section 5 we describe and discuss evaluation for the question answering task on a few datasets that were collected for this research.

2 Related work

2.1 Discourse and IR

Typically, every part in most coherent text has some plausible reason for its presence, some function that it performs to the overall semantics of the text. Rhetorical relations, e.g. *contrast*, *cause*, *explanation*, describe how the parts of a text are linked to each other. Rhetorical relations indicate the different ways in which the parts of a text are linked to each other to form a coherent whole.

Marir and Haouam [17] introduced a thematic relationship between parts of text using RST based on cue phrases to determine the set of rhetorical relations. Once these structures are determined, they are put in an index, which can then be searched not only by keywords, as traditional information retrieval systems do, but also by rhetorical relations.

It was observed [13] that different rhetorical relations perform differently across evaluation measures and query sets. The four rhetorical relations that improve per-

formance over the baseline consistently for all evaluation measures and query sets are: background, cause-result, condition and topic-comment. Topic-comment is one of the overall best-performing rhetorical relations, which in simple terms means that boosting the weight of the topical part of a document improves its estimation of relevance. Regretfully these relations are relatively rare.

Sun and Chai [14] investigated the role of discourse processing and its implication on query expansion for a sequence of questions in scenario-based context Q/A. They consider a sequence of questions as a mini discourse. An empirical examination of three discourse theoretic models indicates that their discourse-based approach can significantly improve Q/A performance over a baseline of plain reference resolution.

In a different task [6] authors parse Web user forum threads to determine the discourse dependencies between posts in order to improve information access over Web forum archives.

Suwandaratna and Perera [16] present a re-ranking approach for Web search that uses discourse structure. They report a heuristic algorithm for refining search results based on their rhetorical relations. Their implementation and evaluation is partly based on a series of ad-hoc choices, making it hard to compare with other approaches. They report a positive user-based evaluation of their system for ten test cases.

Since rhetoric parsers for English [19, 21] has become more available and accurate, their application in search engine indexing is becoming more feasible. As precision and recall of search systems ignoring discourse level information deteriorates, users do not find products, services and information they need, leveraging of linguistic technologies including discourse become realistic for industrial systems. It was shown that discourse features are valuable for passage re-ranking [18]. DTs have been also found to assist in answer indexing to make search more relevant: query keyword should occur in nucleus rather than a satellite of a rhetoric relation [10].

2.2 Discourse Entities

At any point in the discourse, some entities are considered more salient than others (occurring in nucleus parts of DTs), and consequently are expected to exhibit different properties. In Centering Theory [3, 28], entity importance determines how they are realized in an utterance, including pronominalized relation between them. In other discourse theories, entity importance can be defined via topicality and cognitive accessibility [4].

Barzilay and Lapata [7] automatically abstracts a text into a set of entity transition sequences and records distributional, syntactic, and referential information about discourse entities. The authors formulated the coherence assessment as a learning task and show that their entity-based representation is well-suited for ranking-based generation and text classification tasks.

Nguyen and Joty [25] presented a local coherence model based on a convolutional neural network that operates over the distributed representation of entity transitions in the grid representation of a text, can model sufficiently long entity transitions and can incorporate entity-specific features without losing generalization power.

Kuyten et al. [27] developed a search engine that leverages the discourse structure in documents to overcome the limitations associated with the bag-of-words document representations in information retrieval. This system does not address the problem of rhetoric coordination between Q and A, but given a Q, this search engine can retrieve both relevant A and individual statements from A that describe some rhetorical relations to the query.

3 Answering Questions via Discourse Trees

3.1 Imaginary Discourse Tree

The baseline requirement for an A to be relevant to Q is that entities (E) of A cover the entities of Q:

$$\text{E-Q} \subseteq \text{E-A}. \quad (1)$$

Naturally some E-A (entities in an answer) are not explicitly mentioned in Q but are needed to provide a recommendation yielded by Q (recipe-type A).

The next step for an A to be good for Q is to follow the logical flow of Q. Since it is hard to establish relations between entities E, being domain dependent, we try to approximate these relations by logical flow of Q and A, expressible in domain-independent terms, such as rhetorical relation. Hence we require a certain correspondence between DT-Q and DT-A, considering additional labels for DT nodes by **entities** (we denote such DT as *EDT*):

$$\text{EDT-Q} \sim \text{EDT-A}. \quad (2)$$

However a common case is that some entities E are not explicitly mentioned in Q but instead are assumed. Moreover, some entities in A used to answer Q do not occur in A but instead more specific or general entities do. How would we know that these more specific entities are indeed addressing issues from Q? We need some external, additional source which we call imaginary EDT-A to establish these relationships.

This source contains the information on inter-relationships between E which is omitted in Q and/or A but is assumed to be known by the peer. For an automated Q/A system, we want to obtain this knowledge at the discourse level:

$$\text{EDT-Q} \sim \text{EDT-A} + \text{imaginary EDT-A}. \quad (3)$$

3.2 Discourse Tree for Answer and Question

We start with a simple example:

Q: *What is an advantage of electric car?*
A: *No need for gas.*

How can search engine figure out that A is a good one for Q? We have an abstract general-sense entity advantage and a regular noun entity car. We need to link explicit

entities in A *{need, gas}*. Fragments of a possible imaginary EDT-A are shown below:

Q: *[When driving the cruise control][the engine will turn off][when I want to accelerate ,][although the check engine light was off .] [I have turned on the ignition][and listen for the engine pump running][to see][if it is building up vacuum .] [Could there be a problem with the brake sensor under the dash ?] [Looks like there could be a little play in the plug.]*

A: [A faulty brake switch can effect the cruise control .] [If it is ,][there should be a code][stored in the engine control module .] [Since it is not an emissions fault ,][the check engine light will not illuminate .] [First of all , watch the tachometer][to see][if engine speed increases 200 rpm][when this happens .] [If it does ,][the torque converter is unlocking transmission .]

We do not need to know the details how this Enablement occurs, we just need evidence that these rhetorical links exist. We could have used semantic linked between entities but for that we would need a domain-specific ontology.

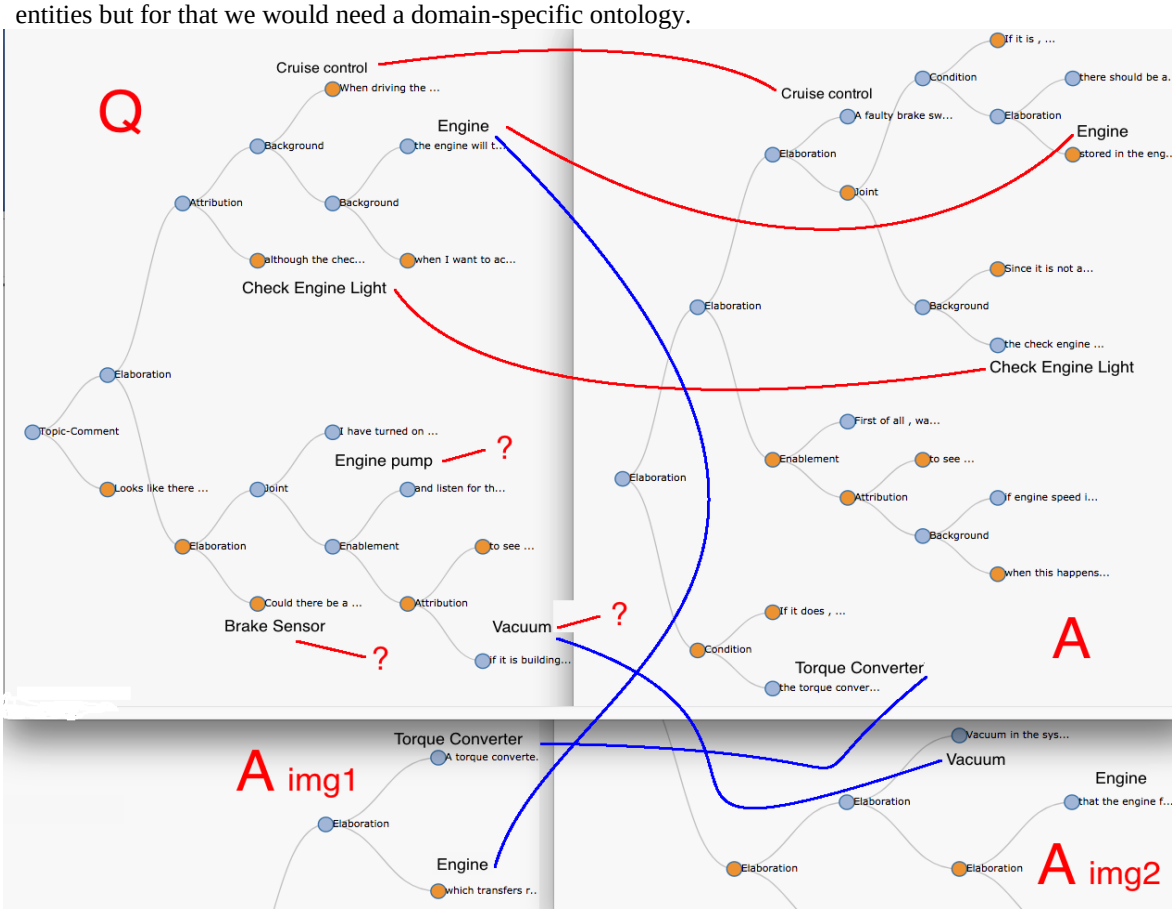

Fig. 1. DTs of *Q*, *A* and imaginary *DT-A*$_{img1}$ and *DT-A* $_{img2}$

Let us explain how a match between a *Q* and an *A* is facilitated by DTs (Fig. 1). *A* explains a situation and also offer some interpretation, as well as recommends a certain course of action. *A* introduces extra entities which are not in *Q*, and needs to involve background knowledge to communicate how they are related to *E-Q*. We do it by setting a correspondence between *E-Q* and *E-A*, shown by the horizontal curly (red) arcs.

Notice that some entities E_0 in Q are un-addressed: they are not mentioned in *A*. E_0-*Q* includes {Engine pump, Brake sensor and Vacuum}. It means that either A is not fully relevant to Q omitting some of its entities E_0 or it uses some other entities instead. Are E_0-*Q* ignored in *A*? To verify the latter possibility, we need to apply some form of background knowledge finding entities E_{img} which are linked to both E_0-*Q* and *E-A*.

It is unclear how *E-A = Torque Convertor* is connected to *Q*. To verify this connection, we obtain a fragment of text from Wikipedia (or another source) about *Torque Convertor*, build *DT-*A_{img1} (shown on the left-bottom of Fig. 1) and observe that it is connected with Engine via rhetoric relation of elaboration. Hence we confirm that *E-A = Torque Convertor* is indeed relevant for *Q* (a vertical blue arc).

Fig. 2. How Imaginary DTs would enable Google search to explain missing keywords

It is also unclear how *E-Q pump* is addressed in *Q*. We find a document on the web about *Engine Pump* and *Vacuum* and attempt to connect them to *E-A*. It turns out that *DT-*A_{img2} connects *Vacuum* and *Engine* via ***elaboration***.

Hence the combined *DT-A* includes real *DT-A* plus *DT-*A_{img1} and *DT-*A_{img2}. Both real and imaginary *DTs* are necessary to demonstrate that an answer is relevant by employing background knowledge in a domain independent manner: no offline ontology construction is required.

Search relevance is then measured as the inverse number of unaddressed E_0 –Q once *DT-A* is augmented with imaginary *DT-*A_{img}. This relevance is then added to a default one.

Fig. 2 shows an example how Imaginary DT component would improve a web search. Currently, search engines show certain keywords they do not identify in a given search result. However, it is possible to indicate how these keywords are relevant to the search result by finding documents where these unidentified keywords are rhetorically connected with the ones occurring in the query. This feature would naturally improve the answer relevance on one hand and provide an explainability for the user on how her keywords are addressed in the answer. In the default search, *munro* is missing. However, by trying to rhetorically connect *munro* with the entities in the question, the Imaginary DT approach finds out that *Munro* is an inventor of automatic transmission. DT fragment is shown with rhetorical relation Attribution, as well as the Wikipedia source for imaginary DT.

4 Question Answering Approach

4.1 Q/A Filtering Algorithm

Given a *Q*, we outline an algorithm that finds the most relevant A such that it has as much of *E-Q* addressed by *E-A*, having a source for imaginary DTs (background knowledge) *B*.

1. Build *EDT-Q*
2. Obtain *E-Q* and form a query for *E-A*
3. Obtain a set of candidate *Answers*
4. For each candidate $A_c \in$ *Answers*:
   (a) Build *DT-*A_c
   (b) Establish mapping *E-Q* $\rightarrow$ *E-*A_c
   (c) Identify E_0 -Q
   (d) Form queries from E_0 –Q and $E_0 - A_c$ (entities which are not in E_0 –Q)
   (e) Obtain search results from *B* for queries and build imaginary *DTs-*A_c
   (f) Calculate the score $|E_0|$ remaining.
5. Select *A* with the best score.

Discourse trees are constructed automatically using state-of-the-art RST-parsers [19, 21].

4.2 Learning on Question Answering Pairs

Besides this algorithm, we outline a machine learning approach to classify <*EDT-Q*, *EDT-A*> pair as correct or incorrect. The training set should include good Q/A pairs and bad Q/A pairs. Therefore a DT-kernel learning approach (SVM TK, [20, 8]) is selected which applies SVM learning to a set of all sub-DTs of the DT for Q/A pair. Tree kernel family of approaches is not very sensitive to errors in parsing (syntactic

and rhetoric) because erroneous sub-trees are mostly random and will unlikely be common among different elements of a training set.

Learning framework is available on our GitHub repository: http://anonymous.4open.science/repository/cfeb10c5-0069-4d9b-adf1-0d1f5086ee00/ (link is anonymized).

5 Experiments

5.1 Data

Traditional Q/A datasets for factoid and non-factoid questions, as well as SemEval and neural Q/A evaluations are not suitable since the questions are shorter and not as complicated to observe a potential contribution of discourse-level analysis. For our evaluation, we formed two convergent Q/A sets:

1. Yahoo! Answer subset [11] of question-answer pairs with broad topics. Out of the set of 140k user questions we selected 3300 of those, which included three to five sentences. Answers for most questions are fairly detailed so no filtering by sentence length was applied to answers.
2. Car repair conversations selected from www.2carpros.com [12] including 9300 Q/A pairs of car problem descriptions vs recommendation on how to rectify them.

For each of these sets, we form the positive one from actual Q/A pairs and the negative one from $Q/A_{\text{similar-entities}}$: $E\text{-}A_{\text{similar-entities}}$ has a strong overlap with *E-A*, although $A_{\text{similar-entities}}$ is not really correct, comprehensive and exact answer. Hence Q/A is reduced to a classification task measured via precision and recall of relating a Q/A pair into a class of correct pairs.

Both of the datasets are available online[1] [2].

5.2 Results

Top two rows in Table 1 show the baseline performance of Q/A and demonstrate that in a complicated domain transition from keyword to matched entities delivers more than 13% performance boost. For the baseline we used standard implementation of Lucene search engine based on matching keywords.

The bottom three rows show the Q/A accuracy when discourse analysis is applied. Assuring a rule-based correspondence between *DT-A* and *DT-Q* gives 13% increase over the baseline, and using imaginary *DT* – further 10%. Finally, proceeding from

[1] http://anonymous.4open.science/repository/cfeb10c5-0069-4d9b-adf1-0d1f5086ee00/examples/CarRepairData_AnswerAnatomyDataset2.csv.zip.
[1] http://anonymous.4open.science/repository/cfeb10c5-0069-4d9b-adf1-0d1f5086ee00/examples/Fidelity_FAQs_AnswerAnatomyDataset1.csv.zip

rule-based to machine learned *Q/A* correspondence (SVM TK) gives the performance gain of about 7%.

The difference between the best performing SVM TK for <*EDT-Q* ∩ *EDT-A+EDT-*A_{imgi}> row and the above row is only the machine learning algorithm: representation is the same.

Table 1. Evaluation results

Source	**Yahoo! Answers**			**Car Repair**		
Search method	**P**	**R**	**F1**	**P**	**R**	**F1**
Baseline (Lucene search engine)	41.8	42.9	42.3	42.5	37.4	39.8
|E-Q ∩ E-A|	53.0	57.8	55.3	54.6	49.3	51.8
|EDT-Q∩EDT-A|	66.3	64.1	65.1	66.8	60.3	63.4
|EDT-Q∩EDT-A +EDT-A_{imgi}|	76.3	78.1	77.2± 3.4	72.1	72.0	72.0±3.6
SVM TK for <EDT-Q, EDT-A +EDT-A_{imgi}>	83.5	82.1	**82.8**±3.1	80.8	78.5	**79.6**±4.1
Human assessment of SVM TK for <EDT-Q∩EDT-A +EDT-A_{imgi}>	81.9	79.7	80.8±7.1	80.3	81.0	80.7±6.8

The bottom row shows the human evaluation of Q/A on a reduced dataset of 200 questions for each domain. We used human evaluation to make sure the way we form the training dataset reflects the Q/A relevance as perceived by humans.

To summarize our experiments, the tree kernel learning of imaginary discourse tree is a preferred approach.

6 Conclusions

Answering questions in the domain of this study is a significantly more complex task than factoid Q/A such as Stanford Q/A database [23], where it is just necessary to involve one or two entities and their parameters. To answer a "how to solve a problem" question, one needs to maintain the logical flow connecting the entities in the questions. Since some entities from Q are inevitably omitted, these would need to be restored from some background knowledge text about these omitted entities and the ones presented in *Q*. Moreover, a logical flow needs to complement that of the *Q*.

Domain-specific ontologies such as the ones related to mechanical problems with cars are very hard and costly to build. In this work we proposed a substitute via do-

main-independent discourse level analysis where we attempt to cover unaddressed parts of DT-A on the fly, finding text fragments in a background knowledge corpus such as Wikipedia. Hence we can do without an ontology that would have to maintain relations between involved entities.

The proposed imaginary DT feature of a Q/A system delivers a substantial increase of accuracy answering complex convergent questions, where it is important to take into account all entities from a question. We observed that relying on rhetoric agreement between Q and A (matching their DTs) improves Q/A accuracy by more than 10% compared to the relevance-focused baseline. Moreover, employing imaginary DTs gives further 10% improvement.

Since we explored the complementarity relation between DT-A and DT-Q and proposed a way to identify imaginary DT-A on demand, the learning feature space is substantially reduced and learning from an available dataset of a limited size such as car repair becomes plausible.

7 References

1. Yangfeng Ji and Noah Smith, A Neural Discourse Structure for Text Categorization. ACL 2017.
2. Alexander Hogenboom, Flavius Frasincar, Franciska de Jong, and Uzay Kaymak. 2015. Using rhetorical structure in sentiment analysis. Communications of the ACM 58(7):69–77.
3. Grosz, Barbara, Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. Computational Linguistics, 21(2):203–225.
4. Gundel, Jaenette K., Nancy Hedberg, and Ron Zacharski. 1993. Cognitive status and the form of referring expressions in discourse. Language 69(2) pp 274-307.
5. William Mann and Sandra Thompson. 1988. Rhetorical Structure Theory: Toward a functional theory of text organization. Text 8(3):243–281.
6. Wang, W., Su, J., Tan, C.L. 2010. Kernel Based Discourse Relation Recognition with Temporal Ordering Information. ACL.
7. Regina Barzilay and Mirella Lapata. 2008. Modeling local coherence: An entity-based approach. Comput. Linguist. 34, 1 (March 2008), 1-34.
8. Boris Galitsky. 2017. Discovering Rhetorical Agreement between a Request and Response. Dialogue & Discourse 8(2) 167-205.
9. Boris Galitsky. 2014. Learning parse structure of paragraphs and its applications in search. Engineering Applications of Artificial Intelligence. 32, 160-84.
10. Boris Galitsky, D. Ilvovsky, D. and S. Kuznetsov. 2015. Rhetoric Map of an Answer to Compound Queries. ACL-2, 681–686.
11. Webscope 2017. Yahoo! Answers Dataset. https://webscope.sandbox.yahoo.com/catalog.php?datatype=l.
12. CarPros 2017. http://www.2carpros.com
13. S. Teufel and M. Moens. 2002. Summarizing scientific articles: Experiments with relevance and rhetorical status. Computational Linguistics, 28(4):409–445.
14. M. Sun and J. Y. Chai. 2007. Discourse processing for context question answering based on linguistic knowledge. Know.-Based Syst., 20:511–526, August 2007.
15. B. Heerschop, F. Goossen, A. Hogenboom, F. Frasincar, U. Kaymak, and F. de Jong. 2011. Polarity analysis of texts using discourse structure. In Proceedings of the 20th ACM

international conference on Information and knowledge management, CIKM '11, pages 1061–1070, New York, NY, USA. ACM.
16. Suwandaratna, N. and U. Perera. 2010. Discourse marker based topic identification and search results refining. In Information and Automation for Sustainability (ICIAFs), 2010 5th International Conference on, pages 119–125.
17. Marir F. and K. Haouam. 2004. Rhetorical structure theory for content-based indexing and retrieval of Web documents, ITRE 2004. 2nd International Conference Information Technology: Research and Education, 2004, pp. 160-164.
18. Jansen, P., M. Surdeanu, and Clark P. 2014. Discourse Complements Lexical Semantics for Nonfactoid Answer Reranking. ACL.
19. Joty, Shafiq R, Giuseppe Carenini, Raymond T Ng, and Yashar Mehdad. 2013. Combining intra-and multi- sentential rhetorical parsing for document-level dis- course analysis. In *ACL (1)*, pages 486–496.
20. Joty, Shafiq R and A. Moschitti. 2014. Discriminative Reranking of Discourse Parses Using Tree Kernels. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).
21. Surdeanu, Mihai, Thomas Hicks, and Marco A. Valenzuela-Escarcega. Two Practical Rhetorical Structure Theory Parsers. Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies: Software Demonstrations (NAACL HLT), 2015.
22. Chali, Y. Shafiq R. Joty, and Sadid A. Hasan. 2009. Complex question answering: unsupervised learning approaches and experiments. J. Artif. Int. Res. 35, 1 (May 2009), 1-47.
23. Pranav Rajpurkar, Zhang, Jian; Lopyrev, Konstantin; Liang, Percy. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. in EMNLP.
24. Christina Lioma, Birger Larsen, and Wei Lu. 2012. Rhetorical relations for information retrieval. SIGIR, Portland, OR, pp. 931-940.
25. Dat Tien Nguyen and Shafiq Joty. 2017. A Neural Local Coherence Model. ACL. pp. 1320-1330.
26. Pranav Rajpurkar, Zhang, Jian; Lopyrev, Konstantin; Liang, Percy. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. in EMNLP.
27. Pascal Kuyten, Danushka Bollegala, Bernd Hollerit, Helmut Prendinger and Kiyoharu Aizawa. 2015. A Discourse Search Engine Based on Rhetorical Structure Theory. Advances in Information Retrieva (ECIR). pp 80--91.
28. Poesio, Massimo, Rosemary Stevenson, Barbara Di Eugenio, and Janet Hitzeman. 2004. Centering: A parametric theory and its instantiations. Computational Linguistics, 30(3):309–363.
29. Anne Louis, Aravind Joshi, Ani Nenkova. 2010. Discourse indicators for content selection in summarization. SIGDIAL, pp. 147–156.