Two Discourse Tree - Based Approaches to Indexing Answers

Boris Galitsky[1] **and Dmitry Ilvovsky**[2]
[1]Oracle Inc. Redwood Shores CA
[2]National Research University Higher School of Economics
boris.galitsky@oracle.com; dilvovsky@hse.ru

Abstract

We explore anatomy of answers with respect to which text fragments from an answer are worth matching with a question and which should not be matched. We apply the Rhetorical Structure Theory to build a discourse tree of an answer and select elementary discourse units that are suitable for indexing. Manual rules for selection of these discourse units as well as automated classification based on web search engine mining are evaluated concerning improving search accuracy. We form two sets of question-answer pairs for FAQ and community QA search domains and use them for evaluation of the proposed indexing methodology, which delivers up to 16 percent improvement in search recall.

1 Introduction

Much online content is available via question-answer pairs such as frequently-asked questions stored on customer portals or internal company portals. Question-answer pairs can be an efficient manner to familiarize a user with content. In some cases, autonomous agents (chatbots) can import such question-answer pairs in order to field user questions.

But such question-answer pairs can contain content that is not central to a topic of an answer. For example, content can include text that is irrelevant or misleading, non-responsive to the particular question, or is neutral and not helpful. If irrelevant text is indexed by a keyword-based search engine, the precision of the search engine is lowered. Moreover, an autonomous agent attempting to answer a user question based on erroneously-indexed text may answer the question incorrectly, resulting in lowered user confidence in the agent. Despite the fact that standard relevance techniques such as ontology, keyword frequency models and separate discourse features (Chali et al., 2009; Jansen et al., 2014) can be applied to solve this problem, a solution is needed for identifying informative parts from all text.

In this paper we propose a new discourse-based approach to determine informative parts of an answer. This approach accesses a body of text including fragments and creates a searchable index including multiple entries, each entry corresponding to a selected fragment. We propose two different methods of fragment selection based on rules and on classification model respectively.

The paper structure is as follows. In Section 2 we introduce the methodology of ***rhetorical anatomy of an answer*** and present an example of that. In Section 3 we propose two Q/A algorithms which is the core part of our approach. In Section 4 we describe and discuss evaluation for the question answering task on a few datasets that were compiled for this research.

2 Rhetoric Anatomy of an Answer

2.1 RST and Discourse Trees

Discourse analysis was proved to be useful in different aspects of question-answering: answer extraction (Zong et al., 2011), modeling rationale in design questions (Kim et al., 2004), query expansion based on relations between sequential questions (Sun and Chai, 2007), etc.

Discourse trees (DT) originate from Rhetorical Structure Theory (RST, Mann and Thompson, 1988). RST models a logical organization of text, relying on relations between parts of text. RST simulates text coherence by forming a hierarchical, connected structure of texts via ***discourse trees***.

Proceedings of Recent Advances in Natural Language Processing, pages 367–372,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/978-954-452-056-4_043

Rhetoric relations are split into the classes of coordinate and subordinate; these relations hold across two or more text spans and therefore implement coherence. These text spans are called elementary discourse units (EDUs). The leaves of a discourse tree correspond to EDUs, the contiguous atomic text spans. Adjacent EDUs are connected by coherence relations (e.g., attribution, sequence), forming higher-level discourse units.

The term "nuclearity" in RST refers to which text segment, fragment, or span, is more central to an author's purpose. A "***nucleus***" refers to a span of text that is more central to an author's purpose than a "***satellite***", which is less central to the topic.

More particularly, we use the determined EDUs of a discourse tree for a body of text and the relations between the EDUs to determine which EDUs should be indexed for search. Different rhetorical relations (e.g., *elaboration*, *contrast*, etc.) can employ different rules.

In general, we hypothesize that a satellite may express a detail of information that is unlikely to be explicitly queried by a user (Galitsky, 2015; Jasinskaja and Karagjosova, 2017).

2.2 Example of Analysis

Let's illustrate our analysis with a question-answer pair and a discourse tree for an answer.

Q: *How should I plan to pay for taxes resulting from converting to a Roth IRA?*
A: *To help maximize your retirement savings, it's generally a good idea to consider not using the proceeds from the conversion to pay the resulting tax costs. Instead, you should consider using cash or other savings held in nonretirement accounts. Using retirement account funds to pay the taxes will reduce the amount you would have available to potentially grow tax-free in your new Roth IRA. Additionally, if you are under 59½, using funds from your retirement account could result in an additional 10% tax penalty, which may significantly reduce the potential benefit of conversion.*

The discourse tree for the answer is shown in Figure 1, and elementary discourse units selected for indexing are circled in green.

The answer could be obtained from a source such as a Frequently Asked Questions (FAQ) database, or a question-answer index. A question-answer index can include multiple questions and corresponding answers. But some fragments in each answer are more informative to answering the corresponding question than other fragments. For example, the phrase "*it is generally a good idea*" adds little to the answer, whereas "*consider not using the proceeds from the conversion*" is informative to the user who posed the original question. Each answer in the question-answer index may provide additional insight in terms of additional questions that can be answered, which are in turn indexed, increasing the usefulness of the data. For example, "*at what age do I pay a penalty for using retirement funds?*" could be answered by the text (e.g., "*age 59 ½*"). We can determine informative text from a body of text and such additional questions that can be answered from the body of text.

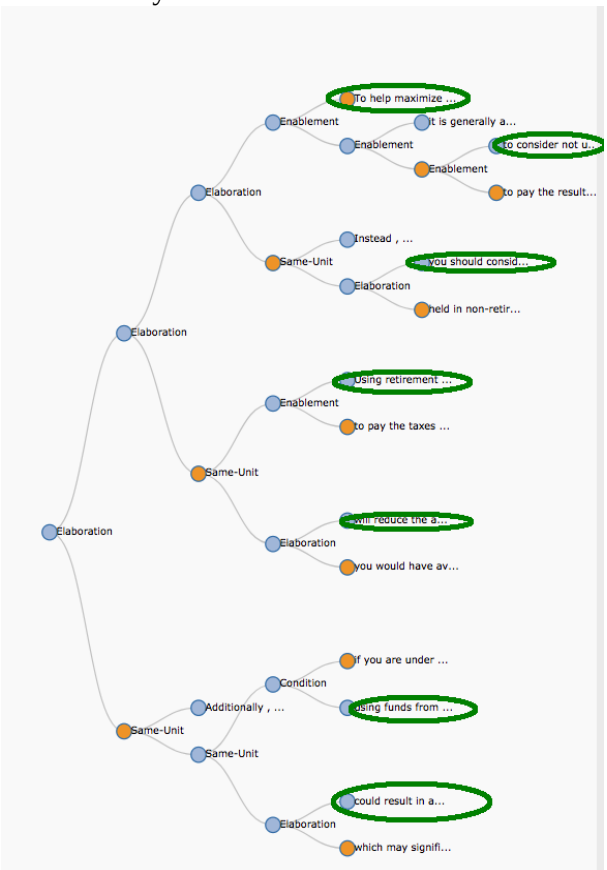

Figure 1: Discourse tree for an answer with the EDUs selected for indexing

2.3 Indexing Rules for Different Rhetorical Relations

The above hypothesis that only EDUs that are ***nucleus*** of rhetoric relations should be indexed and all ***satellite*** EDUs should not be selected for indexing is illustrated by the "*elaboration*" relationship where the nucleus expresses more important information than satellite. But the general rule described above can be subject to certain exceptions. For example, under certain conditions, the "*contrast*" relation can require indexing of the satellite rather than the nucleus. Additionally,

for the "*same-unit*" and "*joint*" relations, both the nucleus and the satellite are indexed. Different rhetoric relations can have different rules, as shown in Table 1 below.

3 The Methodology of Question Answering

The developed methodology of the DT-based analysis of answers is going to be applied in the following way, given an index of Q/A pairs:

1. Search a user query against an index of available questions;
2. If no or too few results, generate additional search queries from the answers indexed by proposed approach;
3. If still no or too few results, search against original answers.

We now focus on 2) and consider two methods for indexing answers: **rule-based** and **classification-based**.

Relation	**Example**	**Indexing rule**
Elaboration	To achieve some state [nucleus] | *do this and that* [satellite]	Nucleus
Enablement	A query may be of the form "*how to achieve some state?*" but less likely be of the form "*what can I achieve doing this and that?*"	Nucleus
Condition	A query may be of the form "*how to achieve some state?*" but less likely of the form "*what can I achieve doing this and that?*"	When the question is of the type "***when/where/under what condition …***", index the ***if*** part (the satellite).
Contrast	Index the nucleus. The satellite includes facts which are unusual, unexpected.	
Same-Unit, Joint	Index both nucleus and satellite because of the symmetric relationship of same-unit.	

Table 1: Indexing rules for rhetorical relations

Once our method construct the indexes, we can build the online search algorithm which combines default functionality provided by the Lucene search engine and syntactic similarity between answer and a query. Search results candidate are selected by Lucene and then matched with the query via finding a maximal common sub-parse tree (Galitsky, 2017).

3.1 Rule-Based Indexing

We take question-answer pairs and create, for each answer, a discourse tree using RST-parser (Surdeanu et al., 2015; Joty et al., 2013). For each non-terminal node in each answer, we then identify a rhetorical relationship associated with the non-terminal node and label each terminal node associated with the non-terminal node as either a ***nucleus*** or a ***satellite***. Then we apply a set of rules (see Table 1) associated with the rhetorical relationships and select, based on the rule, one or more of the fragment associated with the nucleus or the fragment associated with the satellite. Finally, we create a ***searchable index*** of additional questions which includes multiple entries corresponding to one of the selected fragments for the answers.

3.2 Classification-Based Indexing

We use machine learning to learn rules such as those depicted in Table 1. A machine learning problem is formulated as a classification problem that classifies EDUs into a first class that is suitable for indexing (i.e., ***informative***) and forming alternative questions for an answer and a second class that is not suitable for indexing (i.e., ***not informative***).

To accumulate training question-answer pairs with marked answers, we ran selection of queries against short texts. Because longer queries are necessary to assure a corresponding match is nontrivial, we used public question-answer Yahoo! Answers dataset (Webscope, 2017). More specifically, questions from this dataset were formed from a first sentence of the dataset and executed as queries by Microsoft Cognitive Services (Bing Search engine API). Search results which are short texts (4-6 sentences) were selected as such texts suitable for parsing and discourse analysis. Matched fragments of these texts were taken as elements of the training set. Such fragments from the top ten or more pages of search result formed a positive dataset, i.e. informative fragments. For the negative dataset, fragments with matched keywords from the set of lower ranked (100-1000+) search results pages were taken, as these results are assumed to be less relevant.

We applied SVM tree kernel learning (Moschitti, 2006; Severyn and Moschitti, 2012) to

train the model since this algorithm is capable to learn directly on a parse tree structure.

4 Datasets and Evaluation

We used a few datasets to evaluate the contribution of our methodology to search quality.

Yahoo! Answer (Webscope, 2017) subset of question-answer pairs with broad topics where main question is a single sentence (possibly, compound) with ten-fifteen keywords. The dataset includes various domains, and domain knowledge coverage is shallow.

Financial questions[1] scraped from Fidelity.com. This dataset demonstrates how search relevance improvement may occur in a vertical domain with reasonable coverage.

Car repair conversations[2] selected from www.2carpros.com including car problem descriptions and recommendation on how to rectify them. These pairs were extracted from dialogues as first and second utterances.

For each search session, we only consider the first results and reject the others. For all these datasets we assume that there is only one correct answer (from the Q/A pair) and the rest of answers are incorrect.

Evaluation results for the proposed methodology are presented in Table 3. Recall of the baseline search is on average 78% including the improvement by 8% by using syntactic generalization on top of Lucene search (not shown). The relevance of this system is determined by many factors and is therefore not very insightful, so we focus at the change in recall (Δ), from this search system to the one extended by the proposed approach.

[1] https://github.com/bgalitsky/relevance-based-on-parse-trees/examples/Fidelity_FAQs_AnswerAnatomyDataset1.csv.zip

[2] https://github.com/bgalitsky/relevance-based-on-parse-trees/examples/CarRepairData_AnswerAnatomyDataset2.csv.zip.

Dataset	**Question/Answer**	**Total #**	**# generated AQ / # sent**	**Avg # words**
Yahoo! Answers	**Q**	3700	5.5	12.3
	A	3700	8.1	124.1
Fidelity	**Q**	500	3.4	6.2
	A	500	6.2	118.0
Car Repair	**Q**	10000	4.2	5.5
	A	10000	7.0	141.3

Table 2: Dataset statistics

The proposed method delivers about 13 % improvements in the recall have the precision almost unaffected, for the Nucleus/Satellite rules. There is a further 3% improvement by using the automated classifier of EDUs. Since the deployment of such classifier in a domain-dependent manner is associated with substantial efforts, it is not necessarily recommended when this 3% improvement in search accuracy is not critical.

We also compare performance of the proposed search on the extended framework derived from SQuAD 2.0 dataset (Rajpurkar et al., 2018) and applied to *why?* and h*ow-to?* questions. Instead of addressing a question to a single Wikipedia text as standard evaluations do, we run them against all text. We use our approach vs neural extractive reading comprehension one and exceed recall of BiDaf (Gardner et al., 2017) and DeepPavlov (Burtsev et al., 2018) by at least 8% with the search engine trained on our corpus (Table 4).

Dataset / Method	**Baseline**		**Nucleus /Satellite rules,** improvement		**Classification-based,** improvement	
	R	P	ΔR, %	ΔP, %	ΔR, %	ΔP, %
Yahoo! Answers	79	74	+12.5	+0.1	+14	-0.04
Fidelity	77	80	+10	-0.1	+6	+0.1
Car Repair	79	81	**+16**	+0.0	**+18**	+0.0

Table 3: Evaluation results for new datasets

Method	Recall	Precision
BiDaf (AllenNLP)	68	71
DeepPavlov	67	72
Rule-based	75	71
Classification-based	**76**	71

Table 4: Evaluation results for SQuAD dataset

5 Conclusions

In the search engines and chat bot industry, whole texts are usually indexed for search. Because of that, frequently irrelevant answers are delivered because their insignificant keywords (the ones providing auxiliary information and not central for the document) were matched. To overcome this well-known problem, only questions from Q/A pairs are indexed, which dramatically decreases the search recall. To address this limitation of indexing, we proposed and evaluated our approach of indexing only those EDUs of text which are determined to be important (and therefore form alternative questions). This substantially improves the recall in applications such as FAQ search where only questions of Q/A pairs are indexed.

Acknowledgements

Section 3 (algorithms of question answering) were written by Dmitry Ilvovsky supported by the Russian Science Foundation under grant 17-11-01294 and performed at National Research University Higher School of Economics, Russia. Section 4 (experimental investigations) was prepared within the framework of the HSE University Basic Research Program and funded by the Russian Academic Excellence Project '5-100'. The rest of the paper were written and performed at Oracle Corp.

References

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.

Galitsky, B, Ilvovsky, D. and Kuznetsov SO. 2015. *Rhetoric Map of an Answer to Compound Queries.* ACL-2, 681–686.

Galitsky, B. 2017. *Matching parse thickets for open domain question answering*. Data & Knowledge Engineering, Volume 107, January 2017, Pages 24-50.

Gardner, M., Grus, J., Neumann, M., Tafjord, O., Dasigi, P., Liu, N.H., Peters, M., Schmitz, M., & Zettlemoyer, L.S. *A Deep Semantic Natural Language Processing Platform*. arXiv:1803.07640.

Jansen, P., M. Surdeanu, and P. Clark. 2014. *Discourse Complements Lexical Semantics for Non-factoid Answer Reranking.* ACL.

Jasinskaja, K., Karagjosova, E. 2017. *Rhetorical Relations: The Companion to Semantics.* Oxford: Wiley.

Joty, Shafiq R, Giuseppe Carenini, Raymond T Ng, and Yashar Mehdad. 2013. *Combining intra-and multi- sentential rhetorical parsing for document-level discourse analysis.* In ACL (1), pages 486–496.

Kim, S., Bracewell, R., Wallace, K. 2004. *From Discourse Analysis to Answering Design Questions*. International Workshop on the Application of Language and Semantic Technologies to support Knowledge Management Processes (EKAW 2004). At: Whittlebury Hall, Northamptonshire, UK.

M. Burtsev, A. Seliverstov, R. Airapetyan, M. Arkhipov, D. Baymurzina, N. Bushkov, O. Gureenkova, T. Khakhulin, Y. Kuratov, D. Kuznetsov, A. Litinsky, V. Logacheva, A. Lymar, V. Malykh, M. Petrov, V. Polulyakh, L. Pugachev, A. Sorokin, M. Vikhreva, M. Zaynutdinov. 2018. *DeepPavlov: Open-Source Library for Dialogue Systems. ACL-System Demonstrations*. p. 122–127.

M. Sun, J. Y. Chai. 2007. *Discourse Processing for Context Question Answering Based on Linguistic Knowledge.* Knowledge-Based Systems 20(6)(6): 511-526

Mann, William and Sandra Thompson. 1988. *Rhetorical structure theory: Towards a functional theory of text organization. Text* - Interdisciplinary Journal for the Study of Discourse, 8(3):243–281.

Moschitti, A. 2006. *Efficient Convolution Kernels for Dependency and Constituent Syntactic Trees*. In Proceedings of the 17th European Conference on Machine Learning, Berlin, Germany.

Pranav Rajpurkar, Robin Jia, Percy Liang. *Know What You Don't Know: Unanswerable Questions for SQuAD*. arxiv.org/abs/1806.03822

Severyn, A. Moschitti, A. 2012. *Fast Support Vector Machines for Convolution Tree Kernels.* Data Mining Knowledge Discovery 25: 325-357.

Surdeanu, M., Thomas Hicks, and Marco A. Valenzuela-Escarcega. 2015. *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.

Webscope 2017. Yahoo! Answers Dataset. https://webscope.sandbox.yahoo.com/catalog.php?datatype=l

Zong, H., Yu, Z., Guo, J., Xian, Y., Li, J. 2011. *An answer extraction method based on discourse structure and rank learning.* 7th International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE).