Controlling Chat Bot Multi-Document Navigation with the Extended Discourse Trees

Dmitry Ilvovsky
Kazan Federal University,
Kazan, Russia
National Research University
Higher School of Economics,
Moscow, Russia
dilvovsky@hse.ru

Alexander Kirillovich
Kazan Federal University,
Kazan, Russia
alik.kirillovich@gmail.com

Boris Galitsky
Oracle Inc.,
Redwood Shores, CA, USA
bgalitsky@hotmail.com

Abstract

In this paper we learn how to manage a dialogue relying on discourse of its utterances. We define extended discourse trees, introduce means to manipulate with them, and outline scenarios of multi-document navigation to extend the abilities of the interactive information retrieval-based chat bot. We also provide evaluation results of the comparison between conventional search and chat bot enriched with the multi-document navigation.

Keywords: Discourse tree, Dialogue management, Rhetoric structure, Linguistic Linked Open Data

1. Introduction

In this paper we extend the abilities of the interactive chat bot initially developed by (Galitsky, Ilvovsky, 2017) and further improved in (Galitsky, 2019; Galitsky et al., 2019; Galitsky and Ilvovsky 2019). In practice, this chat bot is oriented to work with English language but our approach is language independent. The approach we introduce in this paper is inspired by an idea of a guided search. One source of it is a search methodology designed to show a user an array of different visual possibilities where a searching user may proceed. This is done instead of just navigating to an end point or a terminal answer. This search feature is not looking at images but rather the way those images have been described by users. As particular descriptors show up with sufficient frequency, the system turns them into the categories and sub-categories that accompany search results. This approach is also referred to as faceted search allowing users to narrow down search results by applying multiple filters (Galitsky et al., 2009; Galitsky and McKenna, 2017).

To provide a systematic navigation means to take a user through content exploration, we intend to build upon discourse trees (DTs) for texts and extend the discourse analysis based on RST (Mann and Thompson, 1988) to the level of a corpus of documents. We believe that knowledge exploration should be driven by navigating a discourse tree built for the whole corpus of relevant content. We refer to such a tree as extended discourse tree (EDT). It is a combination of discourse trees of individual paragraphs first across paragraphs in a document and then across documents.

A search engine does not provide a means to navigate through content: it is retained for a search user. Instead, search engine builds an inverse index so that for each query keywords it stores information which paragraph of which document these keywords occur in. Therefore, once a query including multiple documents is submitted, the search engine knows which paragraphs in which documents it should take a search user to.

Most chat bots are designed to imitate human intellectual activity maintaining a dialogue. They try to build a plausible sequence of words to serve as an automated response to user query. Instead, we focus on a chat bot that helps a user to navigate to the exact, professionally written answer as fast as possible.

In addition to narrowing down, zooming into a certain piece of content as search engines do, a chat bot is expected to provide navigational means for content exploration. Therefore, we extend the notion of search inverse index to the one not only allowing to zoom in based on keywords but also on drilling in / drilling out / drilling back in, based on how documents are interconnected.

2. Dialogue Management Approach

2.1. Controlling Chat Bot Navigating with the Extended Discourse Tree

To control the chat bot navigation in a general case, beyond clarification scenarios, we introduce the notion of an extended discourse tree. A conventional discourse tree expresses the author flow of thoughts at the level of paragraph or multiple paragraphs. Conventional discourse tree becomes fairly inaccurate when applied to larger text fragments, or documents. Hence, we extend the notion of a linguistic discourse tree towards an extended discourse tree, a representation for the set of inter-connected documents covering a topic. For a given paragraph, a DT is automatically built by the discourse parser (Joty et al., 2014). We then combine DTs for the paragraphs of documents to the EDT, which is a basis of an interactive content exploration facilitated by the chat bot. We apply structured learning of extended DTs to differentiate between good, cognitively plausible scenarios and counter-intuitive, non-cohesive ones. To provide cohesive answers, we use a measure of rhetorical agreement between a question and an answer by tree kernel learning of their discourse trees (Galitsky and Ilvovsky, 2017).

On the web, information is usually represented in web pages and documents, with certain section structure. Answering questions, forming topics of candidate answers and attempting to provide an answer based on user selected topic are the operations which can be represented with the help of a structure that includes the DTs of texts involved. When a certain portion of text is suggested to a user as an answer, this user might want to drill in something more specific, ascend to a more general level of knowledge or make a side move to a topic at the same level. These user intents of navigating from one portion of text to another can be represented as coordinate or subordinate discourse relations between these portions.

We merge the links between logical parts of paragraphs and the links between documents (Fig. 1). If at the current step the user is interested in drilling in, we navigate her through an Elaboration relation from nucleus to satellite within a paragraph or Elaboration hyperlink to a more specific document. Conversely, if a user decides that the suggested topic is not exactly what he is looking for and wants to return a higher-level view, the system navigates Elaboration relation in the inverse order from satellite to nucleus at either paragraph or intra-document level. The other navigation option is relying on Contrast or Condition relations exploring controversial topics (these rhetorical relations need to be recognized for inter-document case).

Navigation starts with the route node of a section that matches the user query most closely. Then the chat bot attempts to build a set of possible topics, possible understanding of user intent. To do that, it extracts phrases from elementary discourse units that are satellites of the route node of the DT. If the user accepts a given topic, the navigation continues along the chosen edge; otherwise, when no topic covers the user interest, the chat bot backtracks the discourse tree and proceeds to the other section (possibly of other documents) which matched the original user query second best. Inter-document and inter-section edges for relations such as Elaboration play similar role in knowledge exploration navigation to the internal edges of a conventional DT.

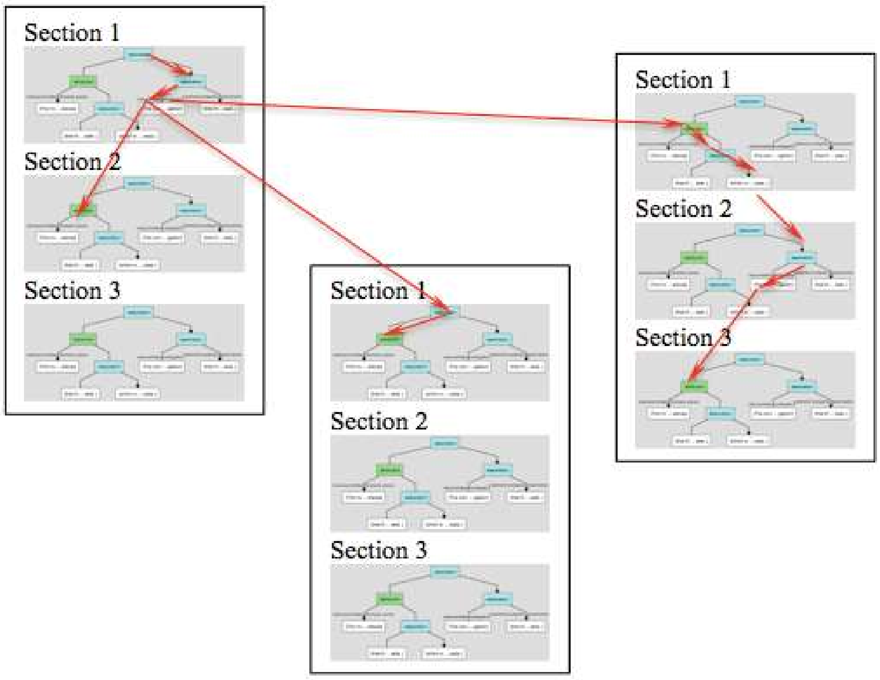

Figure 1: Illustration for the idea of extended DT: intra-paragraph rhetorical relations are combined with inter-document links also labelled as rhetorical relations

2.2. Constructing EDT

To construct EDT, the focus is on building rhetorical links between text fragments (called *elementary discourse units*, or EDU) in different paragraphs or documents. The main difficulty here is to identify a relationship between mentions. The other difficulty is to label an inter-document rhetorical relation. To address it, we form a fictitious text fragment from the respective text fragments of the original paragraph and perform coreferential analysis and discourse parsing.

The input of the EDT algorithm is a set of documents, and an output is an EDT that is encoded as a regular DT with the labels of document identification for each node. The processing flow is as follows:

1. Building a set of all DTs for each paragraph in each document *DTA*;
2. Iterate through all pairs of DT_i and $DT_j \in DTA$;
3. Identify noun phrases and named entities in DT_i and DT_j;
4. Compute overlap and identify common entities E_{ij} between DT_i and DT_j;
5. Establish relationships between occurrences of entities in $E_{i\,j}$ such as *equals, sub-entity, part-of*;
6. Confirm these relationships by forming text fragment merging EDU(E_i) and EDU(E_j) and applying coreference resolution;
7. Form an inter-paragraph rhetorical links R(E_{ij}) for each entity pair occurrence in E_{ij};
8. Classify rhetorical relation for each rhetorical link by forming a text fragment merging EDU(E_i) and EDU(E_j), building its DT and using recognized relation label for this rhetorical link.

To construct conventional DTs, we used existing discourse parser (Joty et al., 2014).

2.3. Example of Navigation

We now present an example of a content exploration scenario based on an extended DT covering three documents (Fig. 2):

Faceted Search
Facets correspond to properties of the information elements. They are often derived by analysis of the text of an item using entity extraction techniques or from pre-existing fields in a database such as author, descriptor, language, and format. Thus, existing web-pages, product descriptions or online collections of articles can be augmented with navigational facets. Within the academic community, faceted search has attracted interest primarily among library and information science researchers, but there is a limited interest of computer science researchers specializing in information retrieval

Entity Extraction
Entity extraction, also known as entity name extraction or named entity recognition, is an information retrieval technique that refers to the process of identifying and classifying key elements from text into pre-defined categories.

Information Retrieval
…

Example 1: Three documents

Exploration scenario is as follows (Fig. 2). Let us imagine that a user is asking a question '*What is faceted search?*'. To understand how it works, this user needs to become fluent with other associated concepts. The chat bot provides further content exploration or search options based on satellite EDUs in the DT of the document '*Faceted search*' (on the top-left). It built multiple DTs (one for each paragraph, two are shown) and formed the following items for content exploration:

- entity extraction;
- information retrieval;
- pre-existing fields in a database;
- augmented with navigational facets.

The user can either follow the link to land on a single piece of information or run a new search to get to multiple search results to choose from. If a user choses 'entity extraction', it is led to the respective document (on the top-right of Fig. 2). The chat bot proceeds to the next iteration, discovering the phrases from satellites of the DT node corresponding to 'entity extraction':

- entity recognition;
- information retrieval.

If a user now selects the second option he would navigate to the 'information retrieval' document.

Whereas a discourse tree of a sentence, paragraph or a document is a well-explored area, algorithms for building a discourse-level representation of a collection of documents in various formats and styles from different sources has not been explored. Irrespectively of the document granularity level, the same relationships such as Elaboration, Contrast and Attribution may hold between the certain portions of text across documents.

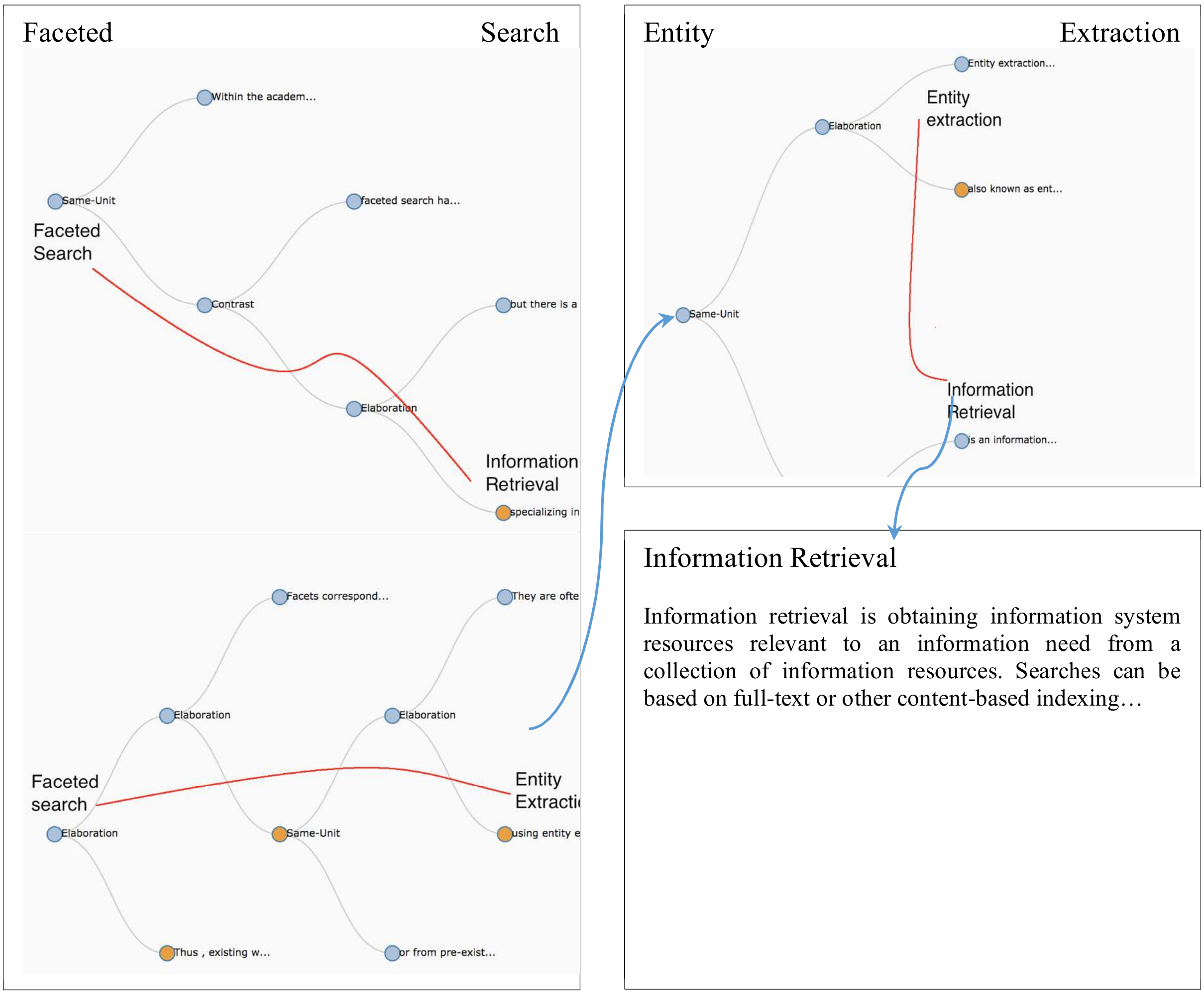

Figure 2: Extended discourse tree for a set of documents used to navigate to a satisfactory answer

3. Evaluation

We compared the efficiency of information access using the proposed chat bot in comparison with a major web search engines such as Google, for the queries where both systems have relevant answers. For search engines, misses are search results preceding the one relevant for a given user. For a chat bot, misses are answers which cause a user to choose other options suggested by the chat bot, or request other topics.

The topics of question included personal finance. Twelve users (author's colleagues) asked the chat bot 15-20 questions reflecting their financial situations, and stopped when they were either satisfied with an answer or dissatisfied and gave up. The same questions were sent to Google, and evaluators had to click on each search results snippet to get the document or a webpage and decide on whether they can be satisfied with it.

The structure of comparison of search efficiency for the chat bot vs the search engine is shown in Fig. 3. The left side of arrows shows that all search results (on the left) are used to form a list of topics for clarification. The arrow on the bottom shows that the bottom answer ended up being selected by the chat bot based on two rounds of user feedback and clarifications. Instead of looking into all search results to find the relevant one (on the left), a user answers a clarification request composed by the chat bot and drills into his topic of interest (on the right). The arrows show how multiple search results on distinct topics are converged to a single clarification request enumerating automatically extracted topics.

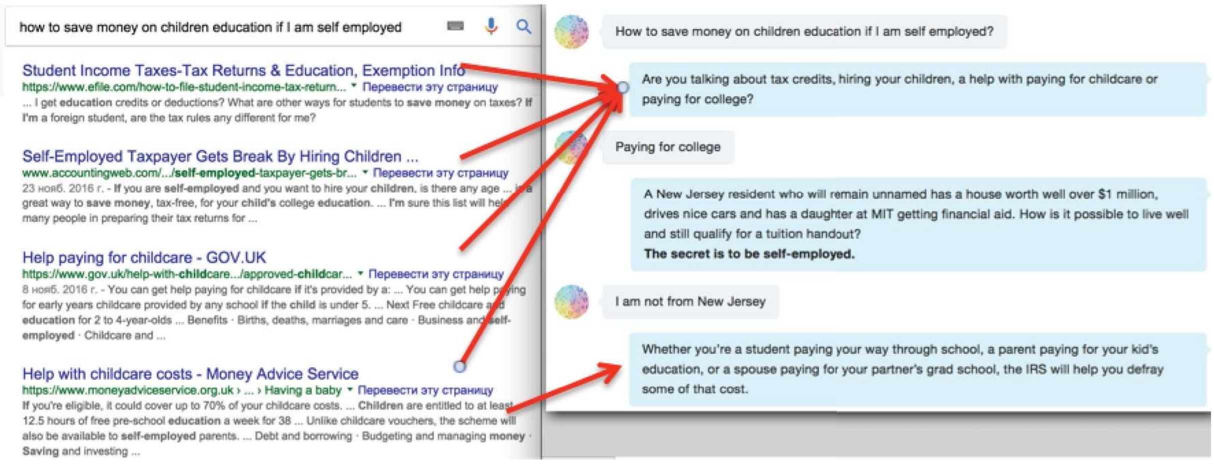

Figure 3: Comparing navigation in a search engine and the chat bot

One can observe (Table 1) that the chat bot time of knowledge exploration session is longer than for the search engine. Although it might seem to be less beneficial for users, businesses prefer users to stay longer on their websites, since the chance of user acquisition grows. Spending 7% more time on reading chat bot answers is expected to allow a user to better familiarize them with a domain, especially when these answers follow the selections of this user. The number of steps of an exploration session for chat bot is a quarter of what is required by a search engine. Traditional ways to measure search engine performance such as MAP and NDCG are also applicable for a comparison between conventional search engines and chat bots with respect to efficiency of information access (Sakai, 2007). We conclude that using a chat bot with extended discourse tree-driven navigation is an efficient and fruitful way of information access, in comparison with conventional search engines and chat bots focused on imitation of a human intellectual activity.

Parameter / search engine	**Conventional web search**	**Chat bot**
Average time to satisfactory search result, sec	45.3	58.1
Average time of unsatisfactory search session (ended in giving up and starting a new search,) sec	65.2	60.5
Average number of iterations to satisfactory search result	5.2	4.4
Average number of iterations to unsatisfactory search result	7.2	5.6

Table 1: Comparison for the chat bot and Google search in the domain of personal finance

4. Related Work

Radev (2000) introduced a cross-document structure theory (CST), a paradigm for multi-document analysis. CST takes into account the rhetorical structure of clusters of related textual documents. He specified taxonomy of relations between documents, cross-document links. CST is intended as a foundation to summarize a collection of documents initiated by a user as well as to navigate it by an abstract information-access machine.

To proceed from RST to CST, one cannot employ the deliberateness of writing style, rely on discourse markers within individual documents. However, it is possible to leverage a logical structure across documents which are systematic, predictable and useful. CST attempts to attach a certain reasoning flow to an imaginary "collective" author of a set of documents.

One of the first studies of rhetorical relations between documents is presented in (Trigg and Weiser, 1987) for scientific papers, such as citation, refutation, revision, equivalence, and comparison. These rhetorical relations are grouped into Normal (inter-document relations) and Commentary (deliberate cross-document relations). However, it is hard to see this model's applicability beyond the scientific domain.

One way to represent the multi-document navigation structure is a multi-document cube. It is a three-dimensional structure that represents related documents with dimensions of *time* (ordered), *source* (unordered) and *position within the document* (ordered).

Discourse disentanglement (such as classification of links between portions of texts or documents) and dialogue/speech/communicative act tagging have been extensively studied (Wang et al., 2011). Discourse disentanglement is the task of splitting a conversation (Elsner and Charniak, 2008) or documents (Wolf and Gibson, 2005) into a sequence of distinct portions of text (sub-discourses). The disentangled discourse is modelled via a tree structure (Grosz and Sidner 1986; Seo et al., 2009), an acyclic graph structure (Rose et al., 1995; Elsner and Charniak, 2008), or a cyclic chain graph structure (Wolf and Gibson, 2005). Speech acts are used to describe the function or role of an utterance in a discourse, similarly to our CDT representation, and have been employed for the analysis of communication means including conversational speech instant messaging, online forums (Kim et al., 2010; Galitsky et al., 2017) and chats (Galitsky and Ilvovsky, 2017). Automated answer scoring benefits from semantic and discourse analyses as well (Wanas et al., 2008). For a more complete review of models for discourse disentanglement and speech act tagging, we refer the reader to (Kim et al., 2010).

Wang et al. (2011) presented the task of parsing user forum threads to determine the labelled dependencies between posts. Three methods, including a dependency parsing approach, are proposed to jointly classify the links (relationships) between posts and the dialogue act (type) of each link. The authors predicted not only the links between posts, but also showed the type of each link, in the form of the discourse structure of the thread.

5. Conclusions and Future Work

We present the first version of a dialogue management system for a chat bot with iterative content exploration that leads a user through a personalized knowledge acquisition session. The chat bot is focused on automated customer support or product recommendation agent that assists a user in learning product features, product usability, suitability, troubleshooting and other related tasks.

The developed dialogue management system is based on the extended discourse trees model. The main contribution of this paper is that it demonstrates applicability of discourse trees in dialog management.

Our current work is undertaken on the following directions:

1) Keeping the topic. In the current version of the system, the chat-bot follows the user's questions, straying off the initial topic. This approach is useful for free conversation systems, but not for task-oriented chat-bots. Currently one of the authors is working on the new approach to dialog management, that tries to avoid digression and keep a user on the main topic of the dialog. We are going to present this new approach at the Dialogue 2020.

2) Linked Open Data integration. In question answering the current version of chat-bot relies only to the data extracted from text documents. Now we are working on complementing these data by the data from Linked Open Data cloud, including domain-independent DBpedia (Lehmann et al., 2015) and our domain-specific mathematical ontology OntoMath^{Edu} (Kirillovich et al., 2020). As an interface between natural language user query and LOD datasets we would rely on the resource from the Linguistic Linked Open Data cloud (Cimiano et al., 2020), such as LLOD representation of WordNet (McCrae et al., 2014), BabelNet (Ehrmann et al., 2014), RuThes (Kirillovich et al., 2017) and FrameNet (Rospocher et al., 2019). We expect that exploitation of LOD cloud can improve user's satisfaction against the baseline obtained in this work.

3) Supporting Russian dialogs. Although the developed approach is language-independent, its actual implementation relies on the discourse parser for English (Joty et.al.,2014) and so now can work only with English texts. We are going to add support for Russian by retraining the parser on the Russian discourse corpus Ru-RSTreebank (Pisarevskaya et al., 2017). In order to achieve interoperability with the parser format, the corpus will be represented in terms of the OLiA Discourse Extensions ontology (Chiarcos, 2014) and integrated to the Linguistic Linked Open Data cloud.

Acknowledgements

The new results presented in this paper were received by D. Ilvovsky and A. Kirillovich during their work in KFU and were funded by Russian Science Foundation according to the research project no. 19-71-10056. These new results are based on the already published discourse models that were being developed by D. Ilvovsky and B. Galitsky in NRU HSE during the previous years.

References

Cimiano, P., Chiarcos, C., McCrae, J.P., and Gracia, J. (2020). Linguistic Linked Open Data Cloud. In Cimiano, P., et al. *Linguistic Linked Data: Representation, Generation and Applications*, pages 29–41. Springer.

Chiarcos, C. (2014). Towards interoperable discourse annotation: Discourse features in the Ontologies of Linguistic Annotation. In Calzolari, N., et al., Eds., *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 4569–4577. ELRA.

Ehrmann, M., Cecconi, F., Vannella, D., McCrae, J., Cimiano, P., and Navigli, R. (2014). Representing Multilingual Data as Linked Data: the Case of BabelNet 2.0. In Calzolari, N., et al., Eds., *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 401–408. ELRA.

Elsner, M. and Charniak, E. (2008). You Talking to Me? A Corpus and Algorithm for Conversation Disentanglement. In Moore, J.D., et al., Eds., *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL-08: HLT)*, pages 834–842. ACL.

Galitsky, B. and Ilvovsky, D. (2017). Chat bot with a Discourse Structure-Driven Dialogue Management. In Martins, A. and Peñas, A., Eds., *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2017 Demo)*, pages 87–90. ACL.

Galitsky, B. and McKenna, E.W. (2017). *Sentiment Extraction from Consumer Reviews for Providing Product Recommendations*. US Patent 9646078B2.

Galitsky, B. (2019). Discourse Level Dialogue Management. In Galitsky, B. *Developing Enterprise Chatbots*, pages 365-426. Springer.

Galitsky, B., González, M.P., and Chesñevar, C.I. (2009). A novel approach for classifying customer complaints through graphs similarities in argumentative dialogue. *Decision Support Systems*, 46(3):717–729.

Galitsky, B. and Ilvovsky, D. (2019). On a Chatbot Conducting Virtual Dialogues. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM 2019)*, pages 2925–2928. ACM.

Galitsky, B, Ilvovsky, D., and Goncharova, E. (2019). On a Chatbot Conducting Dialogue-in-Dialogue. In Nakamura, S., et al., Eds., *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL 2019)*, pages 118–121. ACL.

Grosz, B.J. and Sidner, C.L. (1986). Attention, intention and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Joty, S.R. and Moschitti, A. (2014). Discriminative Reranking of Discourse Parses Using Tree Kernels. In Moschitti, A., et al., Eds., *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 2049–2060. ACL.

Kim, S.N., Wang, L., and Baldwin, T. (2010). Tagging and Linking Web Forum Posts. In Lapata, M. and Sarkar, A., Eds., *Proceedings of the 14th Conference on Computational Natural Language Learning (CoNLL 2010)*, pages 192–202. ACL.

Kirillovich, A., Nevzorova, O., Falileeva, M., Lipachev, E., and Shakirova, L. (2020). OntoMath$^{\mathrm{Edu}}$: a New Linguistically Grounded Educational Mathematical Ontology. In Benzmüller, C. and Miller, B., Eds., *Proceedings of the 13th International Conference on Intelligent Computer Mathematics (CICM 2020)*. Lecture Notes in Artificial Intelligence, vol. 12236. Springer (forthcoming).

Kirillovich, A., Nevzorova, O., Gimadiev. E., and Loukachevitch, N. (2017). RuThes Cloud: Towards a Multilevel Linguistic Linked Open Data Resource for Russian. In Różewski, P. and Lange, C., Eds., *Proceedings of the 8th International Conference on Knowledge Engineering and Semantic Web (KESW 2017)*. Communications in Computer and Information Science, vol. 786, pages. 38–52. Springer, Cham.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., and Bizer, C. (2015). DBpedia: A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*, 6(2):167–195.

Mann, W. and Thompson, S. (1988). Rhetorical Structure Theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

McCrae, J. P., Fellbaum, C., and Cimiano, P. (2014). Publishing and Linking WordNet using lemon and RDF. In Chiarcos, C., et al., Eds., *Proceedings of the 3rd Workshop on Linked Data in Linguistics (LDL-2014)*, pages. 13–16. ELRA.

Pisarevskaya, D., Ananyeva, M., Kobozeva, M., Nasedkin, A., Nikiforova, S., Pavlova, I., and Shelepov, A. (2017). Towards building a discourse-annotated corpus of Russian. In *Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference "Dialogue"*, volume 1, pages 201–212.

Radev, D.R. (2000). A common theory of information fusion from multiple text sources step one: cross-document structure. In Dybkjær, L., et al., Eds., *Proceedings of the 1st SIGdial workshop on Discourse and dialogue (SIGDIAL '00)*, pages 74–83. ACL.

Rose, C.P, Di Eugenio, B., Levin, L.S., and Van Ess-Dykema, C. (1995). Discourse processing of dialogues with multiple threads. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL '95)*, pages 31–38. ACL.

Rospocher, M., Corcoglioniti, F., and Palmero Aprosio, A. (2019). PreMOn: LODifing linguistic predicate models. *Language Resources and Evaluation*, 53:499–524.

Sakai, T. (2007). Alternatives to Bpref. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*, pages 71–78. ACM.

Seo, J., Croft, W.B. and Smith, D.A. (2009). Online community search using thread structure. In *Proceedings of the 18th ACM conference on Information and knowledge management (CIKM '09)*, pages 1907–1910. ACM.

Trigg, R.H. and Weiser, M. (1987). TEXTNET: A network-based approach to text handling. *ACM Transactions on Office Information Systems*, 4(1):1–23.

Wanas, N., El-Saban, M., Ashour, H., and Ammar, W. (2008). Automatic scoring of online discussion posts. In *Proceeding of the 2nd ACM workshop on Information credibility on the web (WICOW '08)*, pages 19–26. ACM.

Wang, L., Lui, M., Kim, S.N., Nivre, J., and Baldwin, T. (2011). Predicting thread discourse structure over technical web forums. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '11)*, pages 13–25. ACM.

Wolf, F. and Gibson, E. (2005). Representing discourse coherence: A corpus-based study. *Computational Linguistics*, 31(2):249–287.