AOC-Poset on discourse and argumentation subgraphs: what can we learn on their dependencies?

Laurine Huber[1], Justine Reynaud[1], Mathilde Dargnat[2], and Yannick Toussaint[1]

[1] LORIA, Université de Lorraine
[2] ATILF, Université de Lorraine and ISC-Marc Jannerod

Abstract. We aim at finding and understanding dependencies between linguistic structures which differ in terms of constraints and expressive power. It has been shown that studying dependencies between the argumentation structure (ARG) and the Rhetorical Structure Theory (RST) is non-trivial and requires a fine methodology. In this paper, we propose to take advantage of the AOC-Poset structure to understand how the subgraphs alignements occur in a small corpus annotated in ARG and RST. We formalize the structures as graphs from which we extract both subgraphs and subgraphs alignments, matching those subgraphs which include the same text segments. Based on these extractions, we build a formal context where the objects are the texts and the attributes are the subgraphs and the subgraphs alignments. We show what we can learn from the dependencies between the structures by mining the AOC-Poset made of these attributes.

Keywords: Formal Concept Analysis · AOC-Poset · Discourse structure · Argumentation structure · Subgraphs alignments

1 Introduction

This paper experiments the use of AOC-Posets (also called Galois sub-hierarchies) to observe and explain how subgraphs coming from two parallel graph-based views of objects are aligned. Formal Concept Analysis and the partial order defined on formal concepts provides a very powerful framework for a fine-grained study of the relations between classes of objects. The experiment concerns a corpus dually annotated as graphs following two distinct theories of discourse and argumentation.

Several linguistic theories aim at annotating the discourse [4,13] or the argumentation [12] structures from texts. Some of these structures may be formalized as graphs where vertices are either segments of text or *artificial* nodes used for structural aspects and directed edges correspond to discourse or argumentative relations. Discovering how a graph built from one theory could be encoded by

ISBN 978–9949–83–557–7, Tallinn University of Technology, 2020.

a graph built from another theory is a real challenge. This could help for comparing their expressive power, exploring their reasoning capabilities and using discourse for predicting the argumentation structure.

In this paper, we build subgraphs alignments from graphs coming from annotations made using two distinct theories.The first one, called Rhetorical Structure Theory[3] [10] (RST) , is used to annotate texts with semantic and pragmatic relations between segments of text (called discourse units). The second one, Argumentation Theory [14] (ARG) is used to annotate texts with argumentative relations between *arguments*, i.e segments of text that have an argumentative function and which are either discourse units or concatenations of adjacent discourse units.

Our corpus of texts has two views that can be represented as ARG and RST graphs, which are then decomposed into subgraphs. An alignment in a text is a pair (S_{RST}, S_{ARG}) of subgraphs, where the S_{RST} subgraph of the RST graph covers the same set of textual segment vertices as S_{ARG}, the subgraph of the ARG graph. If these alignments are frequent in the corpus, i.e they occur in more texts than a given threshold, it would highlight dependencies between the theories. However, alignments are rare and we are interested in understanding in depth which parts of the graphs are aligned, and what the "elements" that prohibits other alignments are. We rely on AOC-Posets, a conceptual structure on object and attribute concepts that has the advantage of being smaller than the full concept-lattice, but still allows one to study dependencies between attributes in terms of subsumption, disjointedness or partial overlap between concepts.

In Section 2, we introduce the graphs built from the two theories and state our problem. In Section 3, we motivate and contextualize our work. In Section 4, we present our methodology and which relevant information on the alignments can be mined from the AOC-poset. Then, we present some results on a small corpus annotated with both ARG and RST and we conclude and present some future research.

2 Problem statement

Our goal is to understand what leads to subgraph alignments in a set of texts (objects) that have two distinct views as graphs, that we decompose into subgraphs (attributes). An *alignment* is a pair of subgraphs (one from the ARG graph, the other one from the RST graph) that cover the same segments of text. We take advantage of AOC-poset for finding dependencies, i.e highlighting alignments that frequently occur in the corpus, and for determining the situations in which alignments occur or do not occur. This section presents the structures on which we are working and states our problem.

[3] Website of the RST: https://www.sfu.ca/rst/

2.1 Textual structures

The corpus from which we extract graphs is made of 112 argumentative texts written to answer a controversial question. For example, the text in Fig. 1 argues about the question *"Should we continue to separate our waste for recycling?"*. Each text has been analysed with two distinct goals: describing discourse and argumentation structures. This led to two distinct annotations, both relying on a single segmentation (in clauses or propositions) but using distinct sets of relations (6 in *ARG*and 26 in *RST*) and distinct constraints. As an example in RST only adjacent segments may be related while there is not such constraint in ARG. This leads to structural differences in the graphs coming from each annotation and thus lack of isomorphism between them.

1. [It's annoying and cumbersome to separate your rubbish properly all the time.]
2. [Three different bin bags stink away in the kitchen
3. and have to be sorted into different wheelie bins.]
4. [But still Germany produces way too much rubbish]
5. [and too many resources are lost
6. when what actually should be separated and recycled is burnt.]
7. [We Berliners should take the chance and become pioneers in waste separation!]

Fig. 1: Segmentation of the text `micro_b001`: discourse segments are in line and argumentative segments are in squared brackets.

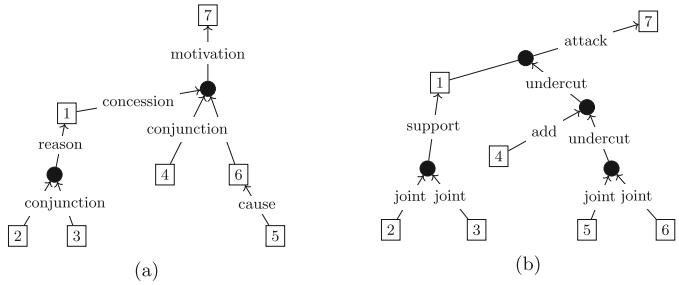

Fig. 2: RST (a) and ARG (b) graphs of the text in Fig. 1. Square nodes are *textual* vertices, and black round nodes are *structural* vertices.

Rhetorical Structure Theory The graphs built in *RST* (see Fig. 2a) aim at describing the intention of the writer about a reader. The annotation using this theory aims at relating adjacent segments of text through discourse relations – represented by labeled directed edges – thus forming bigger segments that are in turn linked to others. To do so, *RST* exploits two distinct types of relations.

Mononuclear relations involve two segments. They are directed, indicating that a segment (the source) is less important than the other (the target) (see segments 5 and 6 in Fig. 2a). Multinuclear relations link two or more segments of equal importance. We introduce *structural* vertices to represent these multinuclear relations in our graphs. They are distinguished from *textual* vertices that represent segments of text. For example, in Fig. 2 vertices 2 and 3 are in a multinuclear `conj` relation: they are thus related through the *structural* vertice to which they are directed. Then, the *structural* vertice is directed to *textual* vertice 1 in a mononuclear `reason` relation.

Argumentation Structure The graphs built in *ARG* (see Fig. 2b) aim at describing how segments are linked by argumentative relations in order to defend a stance. The annotation procedure proposed by [12] uses 5 argumentative relations that are represented as labeled directed edges. However, (1) segments involved in argumentative relations (argumentative segments) may be bigger than segments involved in *RST* relations and (2) *ARG* uses a specific relation that targets a relation instead of a segment. To represent those specific cases, we also use *structural* vertices. Segments 2 and 3 in Fig. 2b are an example of (1). They converge toward a *structural* vertex that is the source of the `support` relation. The *structural* node between segments 1 and 7 is an example of (2). It indicates that the `attack` relation is targeted by another relation.

We thus formalize both *ARG* or *RST* structures as graphs:

Definition 1. *A RST or ARG graph is a labeled directed graph* $G = (V, E)$. $V = V_t \cup V_s$ *is the set of vertices where* V_t *are the textual vertices and* V_s *are the structural vertices.* E *is the set of directed edges labeled by the relations coming from the theories.*

2.2 Finding and understanding alignments

We want to observe dependencies between (parts of) graphs coming from *ARG* and *RST* views. To do so, we observe if alignments between subgraphs are frequent in the sets of graphs, i.e if pairs of subgraphs that cover the same set of textual vertices are occurring frequently in the sets of graphs. For example, in Fig. 1 subgraphs S_A and S_R cover the same textual vertices set $\{1, 2, 3\}$ and thus are aligned. If this alignment was frequent in the corpus, we would be able to interpret it as a dependency between S_A and S_R and further as a dependency between parts of the theories. However, due to the diversity in the annotations and the constraints imposed by the theories, these strict dependencies doesn't occur. We are thus interested in dependencies of other types, for example, (1) if $S1_A$ is aligned to $S1_R$ in some texts but to $S2_R$ in some others, or (2) if $S1_A$ is aligned to $S1_R$ in some texts, to $S2_R$ in some others, but $S1_R$ is aligned to $S1_A$ in some texts and to $S2_A$ in some others.

3 Motivations and related work

Accuosto et al. [1] annotated texts by argumentation structures and used transfer learning for building a model that leverages with features learned from discourse parsing. Compared to models which do not use discourse, results got improved. This led to the idea that discourse structure could help in the task of argumentation mining. To better understand in what ways, it is interesting to clearly establish if discourse and argumentation structures share similarities.

In [11], the authors proposed a corpus where texts are annotated in both argumentation and discourse, in order to study their dependencies. They represented both annotations as (ARG and RST) graphs and did a first empirical study of the overlapping relations between graphs from each theory.

However, two graphs built on one text are usually not isomorphic and the sets of relations used to label the edges are different in size (6 in ARG and 26 in RST). It could thus be interesting to go deeper in the analysis by considering alignments occurring at subgraph level.

For a more systematic comparison and for considering subgraph alignments instead of individual edge alignments, authors in [8] extracted all subgraphs from ARG and RST views and used Redescription Mining [5] for aligning them. Despite promising results, several improvements can be done. The algorithm for extracting redescriptions relies on statistical heuristics that degrade results when working with a small dataset like the one they used (only 112 objects). Also, their approach extracted subgraphs from each view but it did not consider if subgraphs covered the same textual vertices. This led to subgraphs that were considered aligned when they were actually not to be aligned.

Formal Concept Analysis (FCA) [6] provides a powerful framework for studying how objects are grouped according to the attributes they have in common. It is thus relevant to use it to understand dependencies between attributes.

We build AOC-Posets which is most of the time much smaller in size than the full lattice. More precisely, it contains at most $|A| + |O|$ concepts, while the complete lattice may have up to $2^{min(|A|,|O|)}$. AOC-Posets has been used as a tool for different tasks. For example, [2] proposed an exploratory search applied to the field of software product engineering, based on local generation of AOC-Posets. This approach is close to ours as they are trying to find similar behaviors from software bricks.

4 Methods

Our methodology aims, in three steps, at representing the corpus as a formal context (see Fig. 3). We proceed at two different levels. First, at the text level, we extract all subgraphs and we build a pair of subgraphs if two subgraphs are aligned in the text, *i.e.* if they cover the same textual vertices. Then, we study the occurrences of alignments at the corpus level. We compare subgraphs and subgraphs alignments between texts: two subgraphs coming from two different texts are identical as long as their structure is identical, regardless of the label of the vertices.

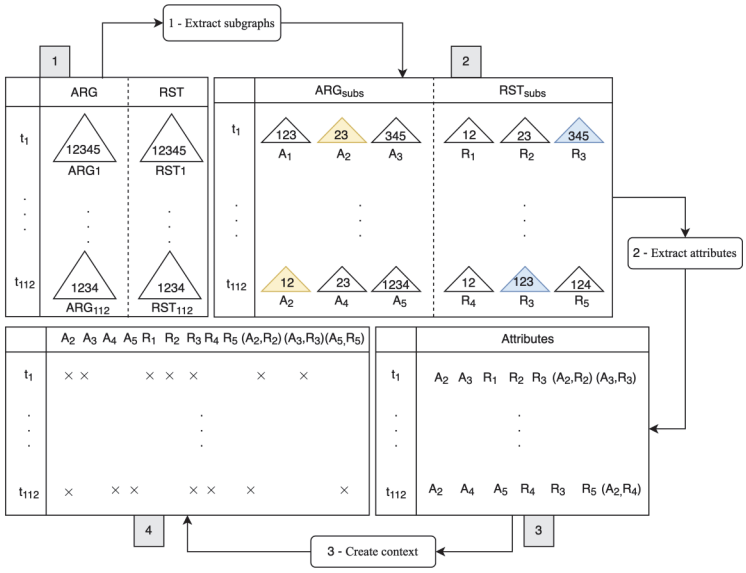

Fig. 3: Overview of the methodology.

4.1 Extracting subgraphs

We introduce two constraints for the extraction of subgraphs to make sure that they involve at least two textual vertices and that they are always weakly connected subgraphs. We call them **valid** subgraphs. For example in Fig. 2a, the ARG subgraph which covers vertices $\{1, 2, 3\}$ is valid while the one covering vertices $\{1, 2, 3, 4\}$ is not.

Definition 2. *A subgraph* $S = (V_t \cup V_s, E_s)$ *is valid iff* S *is weakly connected and if* $|V_t| >= 2$.

For each text $t_i \in T$ we extract all valid subgraphs of ARG_i and RST_i. We obtain 2 sets of subgraphs. In the second table of Fig. 3, triangles represent valid subgraphs and numbers inside represent the nodes covered by it.

$$ARG_{subs} = \{A_x \mid A_x \text{ is a valid subgraph of an } RST \text{ graph}\} \quad (1)$$
$$RST_{subs} = \{R_y \mid R_y \text{ is a valid subgraph of an } ARG \text{ graph}\} \quad (2)$$

Subgraphs that have an identical structure, *ie* that are similar subgraphs regardless of the vertices labels, have a similar label A_x or R_y assigned. For example in the second table of Fig. 3, the two yellow triangles represent subgraphs

that have an identical structure, even if they cover different sets of vertices. The label A_2 is thus assigned to each subgraph. These labeling of subgraphs serves as the basis for constructing the set of attributes.

4.2 Defining attributes of the context

From the sets of subgraphs obtained, we extract pairs of subgraphs (A_x, R_y) that are **aligned** in a text, i.e that cover same textual vertices. For example, in Fig. 2a, the ARG and RST subgraphs which covers the textual vertices $\{7, 4, 6\}$ are aligned because they both cover the same vertices.

Definition 3. *Given* $S_1 = (V_{t1} \cup V_{s1}, E_1)$ *a subgraph of* ARG_{subs} *and* $S_2 = (V_{t2} \cup V_{s2}, E_2)$ *a subgraph of* RST_{subs}, S_1 *and* S_2 *are aligned in a text* t *iff* $V_{t1} = V_{t2}$.

On $ARG_{subs} \times RST_{subs}$, we define the subset of subgraphs that are aligned as follows:

$$AR = \{(A_x, R_y) \subseteq ARG_{subs} \times RST_{subs} \mid \exists t \in T, A_x \text{ and } R_y \text{ are aligned in } t\} \tag{3}$$

and the set of graphs singleton that are aligned as:

$$A = \{A_x \in ARG_{subs} | \exists R_y \in RST_{subs}, (A_x, R_y) \in AR\} \tag{4}$$

$$R = \{R_y \in RST_{subs} | \exists A_x \in ARG_{subs}, (A_x, R_y) \in AR\} \tag{5}$$

Thus, a given text t may contain as an attribute a singleton A_x, and a singleton R_y. If the two subgraphs are aligned, it will also have the attribute (A_x, R_y). For example in Fig. 3, A_2 is aligned with R_2 in t_1 because they both cover vertices 2,3, a pair (A_2, R_2) is thus introduced as an attribute for t_1. In t_{112} however, A_2 is aligned with R_4 and the attribute (A_2, R_4) is thus introduced for this text.

4.3 Creating the context

We determine a formal context K, on the basis of the attributes extracted and built from the graphs. *K := (G,M,I)* where $G = T$ is the set of objets (the texts) and $M = \{A \cup R \cup AR\}$ is the set of attributes (the singleton subgraphs and the aligned pairs) and $(g, m) \in I$ means that a text (*an object*) $g \in G$ has a subgraph and/or a subgraph alignment.

4.4 Taking advantage of AOC-Poset

Explaining alignments among two sets of graphs can be seen as the problem of finding subgraphs that co-occur in a set of objects described by them. Formal Concept Analysis (FCA)[6] is a relevant method to do that. FCA uses two

derivation operators $(.)' : 2^G \mapsto 2^M$ and $(.)' : 2^M \mapsto 2^G$ to build a set of formal concepts from a context. They are defined as $G' = \{m \in M | \forall g \in G, (g, m) \in I\}$ and $M' = \{g \in G | \forall m \in M, (g, m) \in I\}$. Thus, a formal concept (A, B) exists if and only if $A \subseteq G$, $B \subseteq M$, $A' = B$, and $A = B'$. A is called its *extent* and B is called its *intent*. The set of all concepts together with the extent set-inclusion order form the concept lattice of the formal context. For two concepts $C_1 = (A_1, B_1)$ and $C_2 = (A_2, B_2)$, C_1 is said to be *smaller* than C_2 if $A_1 \subset A_2$ and we write $C_1 < C_2$.

The AOC-Poset structure relies on the partial order on the concepts introducing at least an object (*object concept*) or an attribute (*attribute concept*). A concept C introduces an object x (resp. attribute y) when x is in the extent of C but not in any concept $C' < C$ (resp. $C' > C$). While the full concept lattice build from K contains 2120 concepts, the AOC-Poset contains only 379 and can thus be used as a smaller alternative. We used the Hermes algorithm [3] to build the AOC-Poset from K.

The order defined over the concepts of the AOC-Poset may help to observe in which way aligned attributes occur in the corpus. Fig. 4 shows some simplified possible cases (in reduced notation) that we may observe: a node represents a concept and its label gives its local intent.

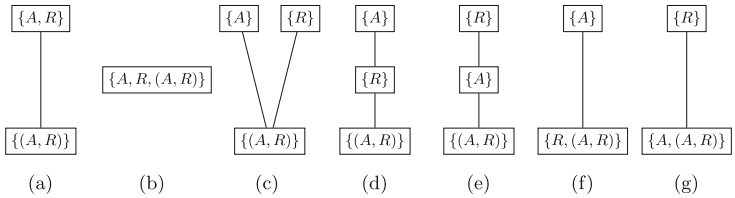

Fig. 4: Substructure of the AOC-Poset: concepts are rectangles labeled by their local intents.

Case 4a in Fig. 4 illustrates that A and R subgraphs may be used in the same texts without being necessarily aligned, contrary to case 4b, where subgraphs are always aligned when they are both used in the same texts. In 4c, the concepts introducing the singletons forming the alignments are incomparable, meaning that there exist some objects having only one of the singletons. Cases 4d (resp. 4e) correspond to cases where, in the texts having an A (resp. R) subgraph, some have also the R (resp. A) subgraph and a subset has both of them aligned. Cases 4f (resp. 4g) illustrates when an A schema is always aligned with an R schema (and thus can be interpreted as one theory that depends on another), and it can be seen as a specialization of the case 4e.

5 Results

5.1 Description of the dataset

The corpus on which we built the formal context is made of 112 texts that we represented with both *ARG* and *RST* graphs having on average 6 textual vertices (min 3, max 13). After building the context with the methodology explained in the previous section, we got 2189 attributes in total, 1278 were singletons that could be aligned and 911 were pairs of aligned subgraphs. These pairs are built from 534 *ARG* subgraphs and 744 *RST* subgraphs. Among them, only 74 have a support greater than 1. The others correspond to the attributes that are specific to a text. We ignore them for simplicity.

x	#ARG subgraphs aligned with x RST subgraphs	#RST subgraphs aligned with x ARG subgraphs
1	407	665
2	57	43
3	28	12
4	10	10
5	11	5
6	3	5
7	4	3
8	1	1
9	2	0
10	3	0
11	0	0

Table 1: Distribution of the # of alignements for each theory

Based on the definitions of the structures and especially the fact that *RST* uses a set of 26 distinct relations compared to 6 in *ARG*, it is likely that a unique *ARG* subgraph may be frequently aligned with several distinct *RST* subgraphs. However, proportions of *ARG* and *RST* alignments roughly followed the same distributions as we can see in Table. 1.

Structure in the AOC-poset (see Fig. 4)	Number	
	support > 1	support > 0
a	0	0
b	0	311
c	67	281
d	0	0
e	0	2
f	2	275
g	5	42
Tot	74	911

Table 2: Structures of the attributes in the AOC-poset for pairs of subgraphs.

5.2 The structure of the alignments in the AOC-Poset

We classified each pair of subgraphs from the attributes set into the classes described in Fig. 4. Table 2 gives the number of attributes for each substructure in the AOC-Poset. We discuss here information learned from the structure of the AOC-Poset on alignments that have a support greater than 1.

Two subgraphs alignments (with a support > 1) correspond to the substructure 4f. Both are in fact introduced in the same concept (c3) (see Fig. 5) meaning that they occur exactly in the same texts. Concepts introducing $a1$ (c1) and $a4$ (c2) form a chain together with c3 while attributes $r128$ and $r132$ are introduced in c1. $a1$ is a subgraph of $a4$, however $a1$ and $a4$ are introduced in different concepts, meaning that $a1$ may be used in a different *"environment"* than the one of $a4$. $r128$ is a subgraph of $r132$, and they are introduced in the same concept so that all texts containing $r128$ also contains $r132$, meaning that the first is never used in another context than the latter. This observation is relevant according to the theories because the multinuclear `list` relation is always used to link at least two textual vertices.

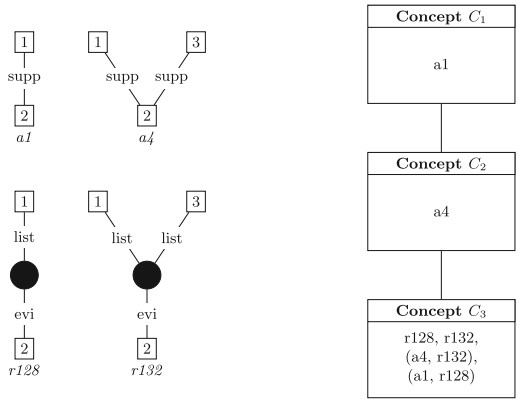

Fig. 5: Subgraphs on the left and their corresponding AOC-Poset substructure on the right.

Four of the five alignments classified as 4e are in fact introduced in the same concept which is also a concept introducing two texts. This attribute-object-concept highlights two texts that were annotated by the exact same structures in both *ARG* and *RST*. The five pairs are introduced in the concept as well as their *ARG* singletons. However, all *RST* attributes are introduced in greater concepts, meaning a dependency of *ARG* with *RST* but not of *RST* with *ARG*.

The substructures classified as 4c are the most frequently observed. Unfortunately, they correspond to cases where we cannot conclude to a unique relation between an ARG subgraph and a RST subgraph. Indeed, most of the time, both subgraphs are implied in more than one alignment, thus implying to study bigger substructure of the AOC-poset, as shown in Figure 6, to make conclusions. These more complex cases are not discussed here due to lack of space.

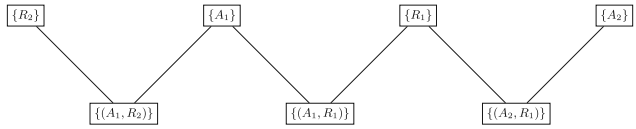

Fig. 6: AOC-Poset Substructure: rectangles show concepts with their local intents.

6 Conclusion and future work

We proposed to approach the problem of finding and understanding alignments in a corpus of texts annotated following two distinct theories. The corpus used and the constraints coming from both theories led to a strong variability in terms of possible alignments between subgraphs. We proposed to represent the corpus as a binary context relating texts with subgraphs or pairs of subgraphs. We used AOC-Poset to highlight specificities in the annotations or generalities that can be used to draw conclusions about the dependencies between the two formalisms.

This work served as a first study on this problem and opens up different possibilities for future work. We found characteristics on the corpus by semi-automatically searching for specific structures on the concepts of the AOC-Poset. This process could be later fully automated allowing a complete understanding of the alignments occurring in the corpus. Some extension of FCA could also be used for this problem. In particular, RCA [9] which allows to consider several types of objects that have their own description and relations with other objects. Instead of considering alignments as pairs of subgraphs that form a specific attribute, we could use RCA to consider alignments as relations between sets of ARG and RST subgraphs. The set of texts would be related to ARG and RST sets with another relation, meaning that a text contains a subgraph. The iterative process allows to integrate relational knowledge in the concept lattices, and would thus highlight new knowledge such as *"texts that have ARG subgraphs a1, a2 also have RST subgraphs r4, r5, and a1 and r4 are aligned."* . It could be interesting to compare this approach with ours, both in terms of complexity and knowledge learned.

Pattern Structures [7] could also be useful as it allows to consider structured data, but for now it provides less tools, in particular for visualization.

7 Acknowledgement

This work was supported partly by the french PIA project "Lorraine Université d'Excellence", reference ANR-15-IDEX-04-LUE.

References

1. Accuosto, P., Saggion, H.: Transferring knowledge from discourse to arguments: A case study with scientific abstracts. In: Proc. of the 6th Workshop on Argument Mining. pp. 41–51. Association for Computational Linguistics (2019)
2. Bazin, A., Carbonnel, J., Kahn, G.: On-demand Generation of AOC-posets: Reducing the Complexity of Conceptual Navigation. In: Foundations of Intelligent Systems - 23rd International Symposium. vol. LNCS, pp. 611–621. Springer (2017)
3. Berry, A., Gutierrez, A., Huchard, M., Napoli, A., Sigayret, A.: Hermes: a simple and efficient algorithm for building the AOC-poset of a binary relation. Annals of Mathematics and Artificial Intelligence **72**(1-2), 45–71 (2014)
4. Busquets, J., Vieu, L., Asher, N.: LA SDRT: Une approche de la cohérence du discours dans la tradition de la sémantique dynamique. Verbum (Presses Universitaires de Nancy) **13**(1), 73–101 (2001)
5. Galbrun, E., Miettinen, P.: From black and white to full color: extending redescription mining outside the Boolean world. Statistical Analysis and Data Mining: The ASA Data Science Journal **5**(4), 284–303 (2012)
6. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundation. Springer-Verlag New York Incorporated (1999)
7. Ganter, B., Kuznetsov, S.O.: Pattern structures and their projections. In: Conceptual Structures: Broadening the Base, 9th International Conference on Conceptual Structures, ICCS 2001, Stanford, CA, USA. pp. 129–142 (2001)
8. Huber, L., Toussaint, Y., Roze, C., Dargnat, M., Braud, C.: Aligning Discourse and Argumentation Structures using Subtrees and Redescription Mining. In: 6th International Workshop on Argument Mining (2019)
9. Huchard, M., Rouane Hacene, A.M., Roume, C., Valtchev, P.: Relational Concept Discovery in Structured Datasets. Annals of Mathematics and Artificial Intelligence **49**(1/4), 39–76 (Apr 2007)
10. Mann, W.C., Thompson, S.A.: Rhetorical structure theory: Toward a functional theory of text organization. Text **8**(3), 243–281 (1988)
11. Musi, E., Stede, M., Kriese, L., Muresan, S., Rocci, A.: A multi-layer annotated corpus of argumentative text: From argument schemes to discourse relations. In: Proc. of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). European Language Resources Association (ELRA), Miyazaki, Japan (2018)
12. Peldszus, A., Stede, M.: From Argument Diagrams to Argumentation Mining in Texts: A Survey. International Journal of Cognitive Informatics and Natural Intelligence (IJCINI) **7**(1), 1–31 (2013)
13. Prasad, R., Miltsakaki, E., Dinesh, N., Lee, A., Joshi, A., Robaldo, L., Webber, B.: The Penn Discourse Treebank 2.0 Annotation Manual. IRCS Technical Reports Series (2007)
14. Stede, M., Afantenos, S., Peldszus, A., Asher, N., Perret, J.: Parallel discourse annotations on a corpus of short texts. In: Proc. of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). Paris, France (2016)