

Mining Arguments From 19th Century Philosophical Texts Using Topic Based Modelling

John Lawrence and Chris Reed
School of Computing,
University of Dundee, UK

Colin Allen
Dept of History & Philosophy of Science,
Indiana University, USA

Simon McAlister and Andrew Ravenscroft
Cass School of Education & Communities,
University of East London, UK

David Bourget
Centre for Digital Philosophy,
University of Western Ontario, Canada

Abstract

In this paper we look at the manual analysis of arguments and how this compares to the current state of automatic argument analysis. These considerations are used to develop a new approach combining a machine learning algorithm to extract propositions from text, with a topic model to determine argument structure. The results of this method are compared to a manual analysis.

1 Introduction

Automatic extraction of meaningful information from natural text remains a major challenge facing computer science and AI. As research on specific tasks in text mining has matured, it has been picked up commercially and enjoyed rapid success. Existing text mining techniques struggle, however, to identify more complex structures in discourse, particularly when they are marked by a complex interplay of surface features rather than simple lexeme choice.

The difficulties in automatically identifying complex structure perhaps suggest why there has been, to date, relatively little work done in the area of argument mining. This stands in contrast to the large number of tools and techniques developed for manual argument analysis.

In this paper we look at the work which has been done to automate argument analysis, as well as considering a range of manual methods. We then apply some of the lessons learnt from these manual approaches to a new argument extraction technique, described in section 3. This technique is applied to a small sample of text extracted from three chapters of “THE ANIMAL MIND: A Text-Book of Comparative Psychology” by Margaret

Floy Washburn, and compared to a high level manual analysis of the same text. We show that despite the small volumes of data considered, this approach can be used to produce, at least, an approximation of the argument structure in a piece of text.

2 Existing Approaches to Extracting Argument from Text

2.1 Manual Argument Analysis

In most cases, manual argument analysis can be split into four distinct stages as shown in Figure 1.

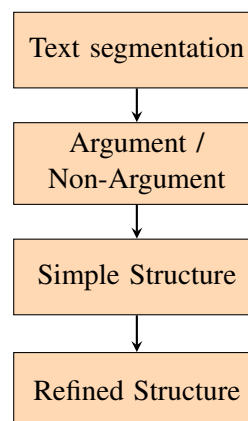


Figure 1: Steps in argument analysis

Text segmentation This involves selecting fragments of text from the original piece that will form the parts of the resulting argument structure. This can often be as simple as highlighting the section of text required, for example in OVA (Bex et al., 2013). Though in some cases, such as the AnalysisWall¹, this is a separate step carried out by a different user.

¹<http://arg.dundee.ac.uk/analysiswall>

Argument / Non-Argument This step involves determining which of the segments previously identified are part of the argument being presented and which are not. For most manual analysis tools this step is performed as an integral part of segmentation: the analyst simply avoids segmenting any parts of the text that are not relevant to the argument. This step can also be performed after determining the argument structure by discarding any segments left unlinked to the rest.

Simple Structure Once the elements of the argument have been determined, the next step is to examine the links between them. This could be as simple as noting segments that are related, but usually includes determining support/attack relations.

Refined Structure Having determined the basic argument structure, some analysis tools allow this to be refined further by adding details such as the argumentation scheme.

2.2 Automatic Argument Analysis

One of the first approaches to argument mining, and perhaps still the most developed, is the work carried out by Moens et al. beginning with (Moens et al., 2007), which attempts to detect the argumentative parts of a text by first splitting the text into sentences and then using features of these sentences to classify each as either “Argument” or “Non-Argument”. This approach was built upon in (Palau and Moens, 2009) where an additional machine learning technique was implemented to classify each Argument sentence as either premise or conclusion.

Although this approach produces reasonable results, with a best accuracy of 76.35% for Argument/Non-Argument classification and f-measures of 68.12% and 74.07% for classification as premise or conclusion, the nature of the technique restricts its usage in a broader context. For example, in general it is possible that a sentence which is not part of an argument in one situation may well be in another. Similarly, a sentence which is a conclusion in one case is often a premise in another.

Another issue with this approach is the original decision to split the text into sentences. While this may work for certain datasets, the problem here is that, in general, multiple propositions often occur

within the same sentence and some parts of a sentence may be part of the argument while others are not.

The work of Moens et al. focused on the first three steps of analysis as mentioned in section 2.1, and this was further developed in (Feng and Hirst, 2011), which looks at fitting one of the top five most common argumentation schemes to an argument that has already undergone successful extraction of conclusions and premises, achieving accuracies of 63-91% for one-against-others classification and 80-94% for pairwise classification.

Despite the limited work carried out on argument mining, there has been significant progress in the related field of opinion mining (Pang and Lee, 2008). This is often performed at the document level, for example to determine whether a product review is positive or negative. Phrase-level sentiment analysis has been performed in a small number of cases, for example (Wilson et al., 2005) where expressions are classified as neutral or polar before determining the polarity of the polar expressions.

Whilst it is clear that sentiment analysis alone cannot give us anything close to the results of manual argument analysis, it is certainly possible that the ability to determine the sentiment of a given expression may help to fine-tune any discovered argument structure.

Another closely related area is Argumentative Zoning (Teufel et al., 1999), where scientific papers are annotated at the sentence level with labels indicating the rhetorical role of the sentence (criticism or support for previous work, comparison of methods, results or goals, etc.). Again, this information could assist in determining structure, and indeed shares some similarities to the topic modelling approach as described in section 3.2 .

3 Methodology

3.1 Text Segmentation

Many existing argument mining approaches, such as (Moens et al., 2007), take a simple approach to text segmentation, for example, simply splitting the input text into sentences, which, as discussed, can lead to problems when generally applied.

There have been some more refined attempts to segment text, combining the segmentation step with Argument/Non-Argument classification. For example, (Madnani et al., 2012) uses three methods: a rule-based system; a supervised probabilis-

tic sequence model; and a principled hybrid version of the two, to separate argumentative discourse into language used to express claims and evidence, and language used to organise them (“shell”). Whilst this approach is instructive, it does not necessarily identify the atomic parts of the argument required for later structural analysis.

The approach that we present here does not consider whether a piece of text is part of an argument, but instead simply aims to split the text into propositions. Proposition segmentation is carried out using a machine learning algorithm to identify boundaries, classifying each word as either the beginning or end of a proposition. Two Naive Bayes classifiers, one to determine the first word of a proposition and one to determine the last, are generated using a set of manually annotated training data. The text given is first split into words and a list of features calculated for each word. The features used are given below:

word The word itself.

length Length of the word.

before The word before.

after The word after. Punctuation is treated as a separate word so, for example, the last word in a sentence may have an after feature of ‘.’.

pos Part of speech as identified by the Python Natural Language Toolkit POS tagger².

Once the classifiers have been trained, these same features can then be determined for each word in the test data and each word can be classified as either ‘start’ or ‘end’. Once the classification has taken place, we run through the text and when a ‘start’ is reached we mark a proposition until the next ‘end’.

3.2 Structure identification

Having extracted propositions from the text we next look at determining the simple structure of the argument being made and attempt to establish links between propositions. We avoid distinguishing between Argument and Non-Argument segments at this stage, instead assuming that any segments left unconnected are after the structure has been identified are Non-Argument.

²<http://www.nltk.org/>

In order to establish these links, we first consider that in many cases an argument can be represented as a tree. This assumption is supported by around 95% of the argument analyses contained in AIFdb (Lawrence et al., 2012) as well as the fact that many manual analysis tools including Araucaria (Reed and Rowe, 2004), iLogos³, Rationale (Van Gelder, 2007) and Carneades (Gordon et al., 2007), limit the user to a tree format.

Furthermore, we assume that the argument tree is generated depth first, specifically that the conclusion is presented first and then a single line of supporting points is followed as far as possible before working back up through the points made. The assumption is grounded in work in computational linguistics that has striven to produce natural-seeming argument structures (Reed and Long, 1997). We aim to be able to construct this tree structure from the text by looking at the topic of each proposition. The idea of relating changes in topic to argument structure is supported by (Cardoso et al., 2013), however, our approach here is the reverse, using changes in topic to deduce the structure, rather than using the structure to find topic boundaries.

Based on these assumptions, we can determine structure by first computing the similarity of each proposition to the others using a Latent Dirichlet Allocation (LDA) model. LDA is a generative model which conforms to a Bayesian inference about the distributions of words in the documents being modelled. Each ‘topic’ in the model is a probability distribution across a set of words from the documents.

To perform the structure identification, a topic model is first generated for the text to be studied and then each proposition identified in the test data is compared to the model, giving a similarity score for each topic. The propositions are then processed in the order in which they appear in the test data. Firstly, the distance between the proposition and its predecessor is calculated as the Euclidean distance between the topic scores. If this is below a set threshold, the proposition is linked to its predecessor. If the threshold is exceeded, the distance is then calculated between the proposition and all the propositions that have come before, if the closest of these is then within a certain distance, an edge is added. If neither of these criteria

³http://www.phil.cmu.edu/projects/argument_mapping/

is met, the proposition is considered unrelated to anything that has gone before.

By adjusting the threshold required to join a proposition to its predecessor we can change how linear the structure is. A higher threshold will increase the chance that a proposition will instead be connected higher up the tree and therefore reduce linearity. The second threshold can be used to alter the connectedness of the resultant structure, with a higher threshold giving more unconnected sections.

It should be noted that the edges obtained do not have any direction, and there is no further detail generated at this stage about the nature of the relation between two linked propositions.

4 Manual Analysis

In order to train and test our automatic analysis approach, we first required some material to be manually analysed. The manual analysis was carried out by an analyst who was familiar with manual analysis techniques, but unaware of the automatic approach that we would be using. In this way we avoided any possibility of fitting the data to the technique. He also chose areas of texts that were established as ‘rich’ in particular topics in animal psychology through the application of the modelling techniques above, the assumption being that these selections would also contain relevant arguments.

The material chosen to be analysed was taken from “THE ANIMAL MIND: A TextBook of Comparative Psychology by Margaret Floy Washburn, 1908” made available to us through the Hathi Trust.

The analyst began with several selected passages from this book and in each case generated an analysis using OVA⁴, an application which links blocks of text using argument nodes. OVA provides a drag-and-drop interface for analysing textual arguments. It is reminiscent of a simplified Araucaria, except that it is designed to work in an online environment, running as an HTML5 canvas application in a browser.

The analyst was instructed only to capture the argument being made in the text as well as they could. Arguments can be mapped at different levels depending upon the choices the analyst prioritises. This is particularly true of volumes such as those analysed here, where, in some cases, the

same topic is pursued for a complete chapter and so there are opportunities to map the extended argument.

In this case the analyst chose to identify discrete semantic passages corresponding to a proposition, albeit one that may be compound. An example is shown in Figure 2. A section of contiguous text from the volume has been segmented and marked up using OVA, where each text box corresponds to such a passage. It is a problem of the era in which the chosen volume is written that there is a verbosity and indirectness of language, so a passage may stretch across several sentences. The content of each box was then edited to contain only argumentative content and a simple structure proposed by linking supporting boxes towards concluding or sub-concluding boxes. Some fifteen OVA maps were constructed to represent the arguments concerned with animal consciousness and with anthropomorphism.

In brief, this analysis approach used OVA as a formal modelling tool, or lens, to characterise and better understand the nature of argument within the texts that were considered, as well as producing a large set of argument maps. Therefore, it represented a data-driven and empirically authentic approach and set of data against which the automated techniques could be considered and compared.

5 Automatic Analysis Results

As discussed in section 4, the manual analysis is at a higher level of abstraction than is carried out in typical approaches to critical thinking and argument analysis (Walton, 2006; Walton et al., 2008), largely because such analysis is very rarely extended to arguments presented at monograph scale (see (Finocchiaro, 1980) for an exception). The manual analysis still, however, represents an ideal to which automatic processing might aspire. In order to train the machine learning algorithms, however, a large dataset of marked propositions is required. To this end, the manual analysis conducted at the higher level is complemented by a more fine-grained analysis of the same text which marks only propositions (and not inter-proposition structure). In this case a proposition was considered to correspond to the smallest span of text containing a single piece of information. It is this detailed analysis of the text which is used as training data for text segmentation.

⁴<http://ova.computing.dundee.ac.uk>

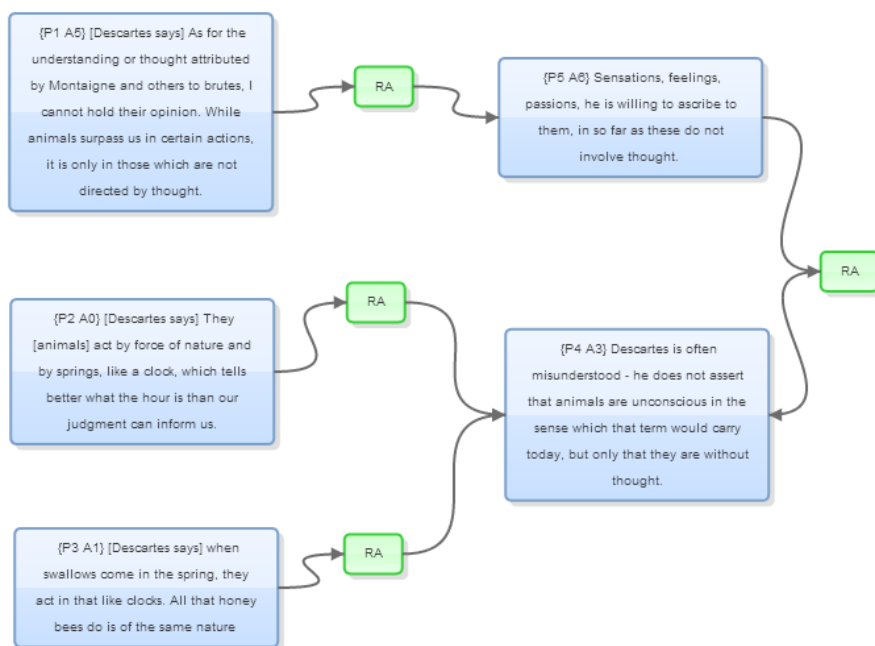


Figure 2: Sample argument map from OVA

5.1 Text segmentation

An obvious place to start, then, is to assess the performance of the proposition identification – that is, using discourse indicators and other surface features as described in section 3.1, to what extent do spans of text automatically extracted match up to spans annotated manually described in section 4? There are four different datasets upon which the algorithms were trained, with each dataset comprising extracted propositions from: (i) raw data directly from Hathi Trust taken only from Chapter 1; (ii) cleaned data (with these errors manually corrected) taken only from Chapter 1; (iii) cleaned data from Chapters 1 and 2; and (iv) cleaned data from Chapters 1, 2 and 4. All the test data is taken from Chapter 1, and in each case the test data was not included in the training dataset.

It is important to establish a base line using the raw text, but it is expected that performance will be poor since randomly interspersed formatting artifacts (such as the title of the chapter as a running header occurring in the middle of a sentence that runs across pages) have a major impact on the surface profile of text spans used by the machine learning algorithms.

The first result to note is the degree of correspondence between the fine-grained propositional analysis (which yielded, in total, around 1,000 propositions) and the corresponding higher level

analysis. As is to be expected, the atomic argument components in the abstract analysis typically cover more than one proposition in the less abstract analysis. In total, however, 88.5% of the propositions marked by the more detailed analysis also appear in the more abstract. That is to say, almost nine-tenths of the material marked as argumentatively relevant in the detailed analysis was also marked as argumentatively relevant in the abstract analysis. This result not only lends confidence to the claim that the two levels are indeed examining the same linguistic phenomena, but also establishes a ‘gold standard’ for the machine learning – given that manual analysis achieves 88.5% correspondence, and it is this analysis which provides the training data, we would not expect the automatic algorithms to be able to perform at a higher level.

Perhaps unsurprisingly, only 11.6% of the propositions automatically extracted from the raw, uncleaned text exactly match spans identified as propositions in the manual analysis. By running the processing on cleaned data, this figure is improved somewhat to 20.0% using training data from Chapter 1 alone. Running the algorithms trained on additional data beyond Chapter 1 yields performance of 17.6% (for Chapters 1 and 2) and 13.9% (for 1, 2 and 4). This dropping off is quite surprising, and points to a lack of homogeneity in

the book as a whole – that is, Chapters 1, 2 and 4 do not provide a strong predictive model for a small subset. This is an important observation, as it suggests the need for careful subsampling for training data. That is, establishing data sets upon which machine learning algorithms can be trained is a highly labour-intensive task. It is vital, therefore, to focus that effort where it will have the most effect. The tailing-off effect witnessed on this dataset suggests that it is more important to subsample ‘horizontally’ across a volume (or set of volumes), taking small extracts from each chapter, rather than subsampling ‘vertically,’ taking larger, more in-depth extracts from fewer places across the volume.

This first set of results is determined using strong matching criteria: that individual propositions must match exactly between automatic and manual analyses. In practice, however, artefacts of the text, including formatting and punctuation, may mean that although a proposition has indeed been identified automatically in the correct way, it is marked as a failure because it is including or excluding a punctuation mark, connective word or other non-propositional material. To allow for this, results were also calculated on the basis of a tolerance of ± 3 words (i.e. space-delimited character strings). On this basis, performance with unformatted text was 17.4% – again, rather poor as is to be expected. With cleaned text, the match rate between manually and artificially marked proposition boundaries was 32.5% for Chapter 1 text alone. Again, performance drops over a larger training dataset (reinforcing the observation above regarding the need for horizontal subsampling), to 26.5% for Chapters 1 and 2, and 25.0% for Chapters 1, 2 and 4.

A further liberal step is to assess automatic proposition identification in terms of argument relevance – i.e. to review the proportion of automatically delimited propositions that are included at all in manual analysis. This then stands in direct comparison to the 88.5% figure mentioned above, representing the proportion of manually identified propositions at a fine-grained level of analysis that are present in amongst the propositions at the coarse-grained level. With unformatted text, the figure is still low at 27.3%, but with cleaned up text, results are much better: for just the text of Chapter 1, the proportion of automatically identified propositions which are included in the man-

ual, coarse-grained analysis is 63.6%, though this drops to 44.4% and 50.0% for training datasets corresponding to Chapters 1 and 2, and to Chapters 1, 2 and 4, respectively. These figures compare favourably with the 88.5% result for human analysis: that is, automatic analysis is relatively good at identifying text spans with argumentative roles.

These results are summarised in Table 1, below. For each of the four datasets, the table lists the proportion of automatically analysed propositions that are identical to those in the (fine-grained level) manual analysis, the proportion that are within three words of the (fine-grained level) manual analysis, and the proportion that are general substrings of the (coarse-grained level) manual analysis (i.e. a measure of argument relevance).

	Identical	± 3Words	Substring
Unformatted	11.6	17.4	27.3
Ch. 1	20.0	32.5	63.6
Ch. 1&2	17.6	26.5	44.4
Ch. 1,2&4	13.9	25.0	50.0

Table 1: Results of automatic proposition processing

5.2 Structure identification

Clearly, identifying the atoms from which argument ‘molecules’ are constructed is only part of the problem: it is also important to recognise the structural relations. Equally clearly, the results described in section 5.1 have plenty of room for improvement in future work. They are, however, strong enough to support further investigation of automatic recognition of structural features (i.e., specifically, features relating to argument structure).

In order to tease out both false positives and false negatives, our analysis here separates precision and recall. Furthermore, all results are given with respect to the coarse-grained analysis of section 4, as no manual structure identification was performed on the fine-grained analysis.

As described in section 3.2, the automatic structure identification currently returns connectedness, not direction (that is, it indicates two argument atoms that are related together in an argument structure, but do not indicate which is premise and which conclusion). The system uses propositional boundaries as input, so can run equally on manually segmented propositions (those used as

training data in section 5.1) or automatically segmented propositions (the results for which were described in Table 1). In the results which follow, we compare performance between manually annotated and automatically extracted propositions. Figures 3 and 4 show sample extracts from the automatic structure recognition algorithms running on manually segmented and automatically segmented propositions respectively.

For all those pairs of (manually or automatically) analysed propositions which the automatic structure recognition algorithms class as being connected, we examine in the manual structural analysis connectedness between propositions in which the text of the analysed propositions appears. Thus, for example, if our analysed propositions are the strings xxx and yyy, and the automatic structure recognition system classes them as connected, we first identify the two propositions (P1 and P2) in the manual analysis which include amongst the text with which they are associated the strings xxx and yyy. Then we check to see if P1 and P2 are (immediately) structurally related. For automatically segmented propositions, precision is 33.3% and recall 50.0%, whilst for manually segmented propositions, precision is 33.3% and recall 18.2%. For automatically extracted propositions, the overlap with the coarse-grained analysis was small – just four propositions – so the results should be treated with some caution. Precision and recall for the manually extracted propositions however is based on a larger dataset (n=26), so the results are disappointing. One reason is that with the manual analysis at a significantly more coarse-grained level, propositions that were identified as being structurally connected were quite often in the same atomic unit in the manual analysis, thus being rejected as a false positive by the analysis engine. As a result, we also consider a more liberal definition of a correctly identified link between propositions, in which success is recorded if either:

(a) for any two manually or automatically analysed propositions (p1, p2) that the automatic structure recognition indicates as connected, there is a structural connection between manually analysed propositions (P1, P2) where p1 is included in P1 and p2 included in P2

or

(b) for any two manually or automatically analysed propositions (p1, p2) that the automatic struc-

ture recognition indicates as connected, there is a single manually analysed propositions (P1) where p1 and p2 are both included in P1

Under this rubric, automatic structure recognition with automatically segmented propositions has precision of 66.6% and recall of 100% (but again, only on a dataset of n=4), and more significantly, automatic structure recognition with manually segmented propositions has precision 72.2% and recall 76.5%. These results are summarised in Table 2.

	Automatically segmented propositions	Manually segmented propositions
In separate propositions	n=4, P=33.3%, R=50.0%	n=26, P=33.3%, R=18.2%
In separate or the same proposition	n=4, P=66.6%, R=100.0%	n=26, P=72.2%, R=76.5%

Table 2: Results of automatic structure generation

The results are encouraging, but larger scale analysis is required to further test the reliability of the extant algorithms.

6 Conclusion

With fewer than one hundred atomic argument components analysed at the coarse-grained level, and barely 1,000 propositions at the fine-grained level, the availability of training data is a major hurdle. Developing these training sets is demanding and extremely labour intensive. One possibility is to increasingly make available and reuse datasets between projects. Infrastructure efforts such as aifdb.org make this more realistic, with around 15,000 analysed propositions in around 1,200 arguments, though as scale increases, quality management (e.g. over crowdsourced contributions) becomes an increasing challenge.

With sustained scholarly input, however, in conjunction with crossproject import and export, we would expect these datasets to increase 10 to 100 fold over the next year or two, which will support rapid expansion in training and test data sets for the next generation of argument mining algorithms.

Despite the lack of training data currently available, we have shown that automatic segmentation of propositions in a text on the basis of relatively simple features at the surface and syntactic levels

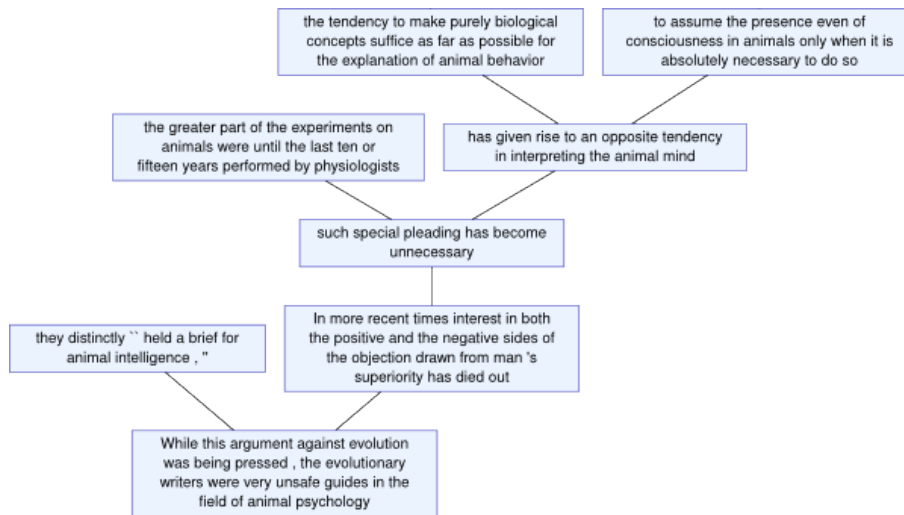


Figure 3: Example of automated structure recognition using manually identified propositions

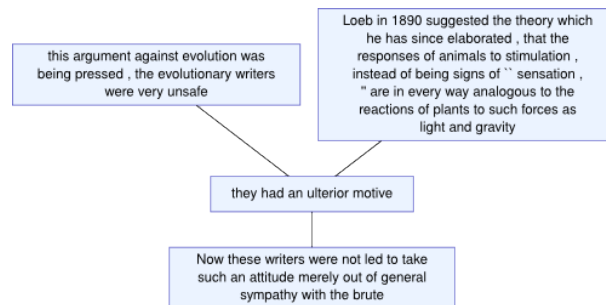


Figure 4: Example of automated structure recognition using automatically identified propositions

is feasible, though generalisation between chapters, volumes and, ultimately, genres, is extremely demanding.

Automatic identification of at least some structural features of argument is surprisingly robust, even at this early stage, though more sophisticated structure such as determining the inferential directionality and inferential type is likely to be much more challenging.

We have also shown that automatic segmentation and automatic structure recognition can be connected to determine at least an approximation of the argument structure in a piece of text, though much more data is required to test its applicability at scale.

6.1 Future Work

Significantly expanded datasets are crucial to further development of these techniques. This will require collaboration amongst analysts as well as the further development of tools for collaborating on and sharing analyses.

Propositional segmentation results could be im-

proved by making more thorough use of syntactic information such as clausal completeness. Combining a range of techniques to determine propositions would counteract weaknesses that each may face individually.

With a significant foundation for argument structure analysis, it is hoped that future work can focus on extending and refining sets of algorithms and heuristics based on both statistical and deep learning mechanisms for exploiting not just topical information, but also the logical, semantic, inferential and dialogical structures latent in argumentative text.

7 Acknowledgements

The authors would like to thank the Digging Into Data challenge funded by JISC in the UK and NEH in the US under project CIINN01, "Digging By Debating" which in part supported the research reported here.

References

- F. Bex, J. Lawrence, M. Snaith, and C.A. Reed. 2013. Implementing the argument web. *Communications of the ACM*, 56(10):56–73.
- P.C. Cardoso, M. Taboada, and T.A. Pardo. 2013. On the contribution of discourse structure to topic segmentation. In *Proceedings of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 92–96. Association for Computational Linguistics.
- V.W. Feng and G. Hirst. 2011. Classifying arguments by scheme. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 987–996. Association for Computational Linguistics.
- Maurice A. Finocchiaro. 1980. Galileo and the art of reasoning. rhetorical foundations of logic and scientific method. *Boston Studies in the Philosophy of Science New York, NY*, 61.
- Thomas F Gordon, Henry Prakken, and Douglas Walton. 2007. The carneades model of argument and burden of proof. *Artificial Intelligence*, 171(10):875–896.
- John Lawrence, Floris Bex, Chris Reed, and Mark Snaith. 2012. Aifdb: Infrastructure for the argument web. In *COMMA*, pages 515–516.
- N. Madnani, M. Heilman, J. Tetreault, and M. Chodorow. 2012. Identifying high-level organizational elements in argumentative discourse. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 20–28. Association for Computational Linguistics.
- M.F. Moens, E. Boiy, R.M. Palau, and C. Reed. 2007. Automatic detection of arguments in legal texts. In *Proceedings of the 11th international conference on Artificial intelligence and law*, pages 225–230. ACM.
- R.M. Palau and M.F. Moens. 2009. Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 98–107. ACM.
- Bo Pang and Lillian Lee. 2008. *Opinion mining and sentiment analysis*. Now Pub.
- Chris Reed and Derek Long. 1997. Content ordering in the generation of persuasive discourse. In *IJCAI (2)*, pages 1022–1029. Morgan Kaufmann.
- Chris Reed and Glenn Rowe. 2004. Araucaria: Software for argument analysis, diagramming and representation. *International Journal on Artificial Intelligence Tools*, 13(04):961–979.
- S. Teufel, J. Carletta, and M. Moens. 1999. An annotation scheme for discourse-level argumentation in research articles. In *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*, pages 110–117. Association for Computational Linguistics.
- Tim Van Gelder. 2007. The rationale for rationale. *Law, probability and risk*, 6(1-4):23–42.
- D Walton, C Reed, and F Macagno. 2008. *Argumentation Schemes*. Cambridge University Press.
- D Walton. 2006. *Fundamentals of critical argumentation*. Cambridge University Press.
- Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics.