Epicurus: a platform for the visualisation of forensic documents based on a linguistic approach

Vassiliki Somaraki
Computing and Engineering
University of Huddersfield
Huddersfield, United Kingdom
v.somaraki@hud.ac.uk

Zhijie Xu
Computing and Engineering
University of Huddersfield
Huddersfield, United Kingdom
z.xu@hud.ac.uk

***Abstract*—This paper presents a tool to visualize a cognitive model of human discourse processing known as Text World Theory (TWT) which is used to facilitate forensic discourse analysis. XML files are designed based on a linguistic annotation scheme. It encompasses the range of descriptive categories defined in TWT. Epicurus is a tool that can parse and visualize those XML files into HTML. The tool is designed for ease of language data annotation and to facilitate evidential analysis by i) visualizing the complex narratives (text-worlds) projected from any given forensic text and ii) reconstructing and visualizing reported events in timeline fashion.**

Keywords-XML,visualisation,TWT,forensic data

I. *INTRODUCTION*

This paper describes a tool that visualizes complex language data using Extensible Markup Language (XML) in order to improve computer based techniques for language data processing pertaining to forensic investigations. Software currently exists for reconstructing and diagramming arguments (e.g. the freeware tool, Araucaria) but current mark-up schemes of argumentation lack both objectivity and linguistic sophistication. Building on the work of the project team (see, for example, [1], [2], [3], [4], [5] and [6]), we are developing a visualisation tool for a mark-up scheme based on Text World Theory, which is a model of human discourse processing developed in Linguistics.

Text World Theory will provide a rigorous and objective means of identifying those elements of a text which are likely to be key to successful visualisation. In developing a method for visualising complex language data we aim to build on and improve existing techniques by operationalising a sophisticated model of human discourse processing known as Text World Theory. We will use Text World Theory to inform the development of a method of mark-up which we will then apply to the data, automating this as far as possible. This mark-up scheme will be a significant improvement on existing practice, since it will provide an objective and replicable method that removes the need for intuitive judgements about which elements of the data are likely to be important for producing computer-based visualisations.

Key to this process of producing visualisations is the ability to discern relationships between pieces of evidence and relate these to specific contexts. The tool is aimed at improving the readability of forensic reports of crime scenes, which typically are lengthy and contain large amount s of complex information which must be distilled by the reader in order for them to be able to assess its value to the case in hand. It is widely acknowledged that well-crafted data visualisation helps uncover trends, realise insights, explore sources, and tell stories. To present the TWT discourse result effectively, both aesthetic form and functionality need to go hand in hand, providing insights into complex data set by communicating its key-aspects in an intuitive way.

II. BACKGROUND

Forensic reports of crime scenes are typically lengthy and contain large amount of complex information. In order to enhance readability the use of visualisation tool is necessary. Such tools could be based on logical rules for legal arguments. In past decades, there has been a large amount of research on the development of logical tools for legal argument (see, e.g., the work of [7], [8], [9], [10], [11], [12] and [13] Argument forms that have been studied include arguments concerning exceptions to rules, conflicts of reasons and rule applicability. The logical tools that have recently been developed can be categorized under three headings: defeasibility, integration of logical levels, and the process character of argument [14].
[15] discussed artificial arguments assistants for defeasible argumentation. He presented two systems ARGUE! based on CUMULA and ARGUMED based on DEFLOG. Argument assistance systems can serve in a context of more than one user: such argument mediation systems can be used to keep track of diverging positions and assist in the evaluation of opinions. More specifically, argument assistance systems are aids to drafting and generating arguments.

The Argue!-system [16] is a system for computer-mediated defeasible argumentation with a graphical user interface. Central actions of defeasible argumentation are inference, justification and attack. Argumentation starts with making a statement .statements can be if two types: assumption and issues. The Argue!-system is an evaluative system for argument mediation: the user provides the argumentation data

such as assumptions, issues, reasons, and attacks. The system determines the justification status of statements i.e., whether they are justified, unjustified, or neither. When the user enters new argumentation data, statements obtain their initial value: assumptions are initially justified, issues are initially neither justified nor unjustified. Justified statements are shown in white boxes, unjustified statements in crossed white boxes, statements that are neither justified nor unjustified in grey boxes. The Argue!-system has two built-in algorithms that help determining the justification status of arguments: 'evaluate' and 'jump'. The system evaluates the statements when the user clicks the 'Evaluate'-button. The system evaluates statements in rounds: the justification statuses of statements are used as input for computing their status in the next. The Argue!-system has the following two evaluation rules:

- If a statement is an assumption, it is justified.
- If a statement is an issue, and has justified support, it is justified.

The second central action of defeasible argumentation is justification. When a new issue is raised it can be justified by giving support for it in the form of an assumption. The third central action of defeasible argumentation is attack. The user has added a defeater, visualized by a special visual shape that consists of two connected rectangles. The argument configuration contained in the first rectangle is challenging the argument configuration in the second is challenged. The defeater represents that the new issue is a counterargument.
Reinstatement is typical for defeasible argumentation. An argument is said to be reinstated if it becomes undefeated after being undefeated.

ARGUMED [17] is the evolution of the ARGUE!. Instead of using forms to enter argumentative data, the user can interact though the screen with the program. With respect to the argumentation theory, attack is no longer limited to undercutting exceptions, but it is possible to attack any statement. Moreover the arrows used to represent support or attack, are considered as conditional statements, which allow a natural treatment of warrants and undercutters. The dialectical arguments consist of statements that can have two types of connections between them: a statement can support another, or a statement can attack another. The former is indicated by a pointed arrow between statements, the latter by an arrow ending in a cross.
In general, dialectical arguments are finite structures that result from a finite number of applications of three kinds of construction types: making a statement; supporting a previously made statement by a reason for it; and, attacking a previously made statement by a reason against it.

The evaluation of dialectical arguments with respect to a set of prima facie justified assumptions is naturally constrained as follows:

- A statement is justified if and only if (a) it is an assumption, against which there is no defeating reason, or (b) it is an issue, for which there is a justifying reason. A statement is defeated if and only if there is a defeating reason against it.
- A reason is justifying if and only if the reason and the conditional underlying the corresponding supporting argument step are justified.
- A reason is defeating if and only if the reason and the conditional underlying the corresponding attacking argument step are justified.

[18] presented a software which aimed to help the diagramming process of argumentation analysis, the Araucaria. The Araucaria system provides an interface which supports the diagramming process, and then saves the result using AML, an open standard, designed in XML, for describing argument structure. Araucaria aims to be of use not only in pedagogical situations, but also in support of research activity. As a result, it has been designed from the outset to handle more advanced argumentation theoretic concepts such as schemes, which capture stereotypical patterns of reasoning. The software is also designed to be compatible with a number of applications, including dialogic interaction and online corpus provision. The assumptions behind Araucaria follow the same pattern: a single text might be analysed in several different ways, depending upon a variety of analytical choices. The judgements concerning the delimitation of argument components can vary, depending upon the aims of the analyst and the clarity of the text itself.

[19] presented EventFlow which is an interactive visual query tool with the task of finding interesting and important event sequences. Although the tool can be used in any kind of data the models was tested against medical data. The tool can be used on point-based events and on interval-based events. It can be used in multiple records and it can display the records either as individual display or as aggregated display. The development of the EventFlow was based on three major spheres: temporal logic, temporal querying and temporal visualisation. This tool provides to the user the following tools to manipulate the display of the events. The aim of those mechanisms is to help the user to reduce the volume of the information displayed on the screen and thus make it easier for the user to understand the data. the main contributions of EventFlow are:

- A visual representation of interval events for both individual and aggregated displays.
- A set of controls for simplifying and exploring records containing interval events.
- A simple visual query language for professionals in nontechnical fields that allows users to specify the presence or absence of both point and interval events.

LifeLines [20] provide a general visualisation environment for personal histories that can be applied to medical and court records, professional histories and other types of biographical data. A one screen overview shows multiple facets of the

records. Aspects, for example medical conditions or legal cases, are displayed as individual time lines, while icons indicate discrete events, such as physician consultations or legal reviews. Line colour and thickness illustrate relationships or significance, rescaling tools and filters allow users to focus on part of the information. LifeLines reduce the chances of missing information, facilitate spotting anomalies and trends, streamline access to details, while remaining tailorable and easily transferable between applications

- Reduce the chances of missing information. Because the data entry is performed over a long period of time by different people the LifeLines overview assists users in reviewing a disparate record. Yet unseen, or recently added and updated information can be revealed by highlighting.
- Facilitate the spotting of anomalies and trends. Intervals are easier to estimate on a timeline than in a table of dates. Repetitions of series of events result in visible patterns.
- Streamline the access to details. LifeLines act as large menus from which large numbers of detail screens can be accessed in a single step
- Remain simple and tailorable to various applications. The long term success of any record format depends on its sharability among collaborating services. LifeLines only uses high level data that can act as reference pointers to other services records.

Lifelines2 is an extension of Lifelines. Lifelines was designed to summarize the entirety of a single personal history record (e.g. a medical record). In contrast, Lifelines2 displays selected subsets of the records. The output of a query (e.g. Find all patients and Partners Health Care) [21]. Each record is vertically stacked on alternating background colour and identified by its ID on the left. For example in the case of the above query in medical records, Asthma and pneumonia diagnosis events appear as coloured triangle icons on the timeline. By default all records are presented using the same absolute time scale (with the corresponding years or month labels displayed at the top) and the display is fitted so that the entire date range fits in the screen.

- LifeFlow [22] developed for the event sequence analysis. LifeFlow is scalable, can summarize all possible sequences, and represents the temporal spacing of the events within sequences. LifeFlow can summarize not only all possible sequences but also the temporal spacing of the events within sequences. LifeFLow is implemented in Java SE 6.0 and includes interaction features to support exploration such as, zooming, tooltipping and non-temporal attributes.

LifeFlow concerns event-based points, for many records and offers the options of individual record display and the aggregated records display.

III. TEXT WORLD THEORY (TWT)

One can argue that after processing crime report using corpus linguistic tools, those reports could be organised into events and therefore could be visualized in the form of timelines. Such organisation of temporal events in time line could enhance spotting important information and focusing on particular points of a story.

Text World Theory makes an initial distinction between the discourse world and the text world. The discourse world is the immediate real-world situation in which a writer communicates with a reader. The text world is a mental representation constructed from the language and the schematic knowledge it evokes. Included within the discourse world is the experience of all participants in the discourse, as well as all surrounding physical objects and entities, and together these form a context. [23] defines context as 'the relevant situational background(s) for and in a particular discourse'. The key word here is 'relevant', since the potential context for any given discourse world is vast. Discourse participants restrict this by only considering common ground information; i.e. only that information which is necessary for the interpretation of the discourse in question.

Participants in the discourse world use the textual and common ground information present within it to construct a text world – i.e. a mental representation of the text. Text worlds are composed of world-building elements and function-advancing propositions, both of which are recovered from the text. World-building elements consist of time (realised through the tense and aspect of verb phrases), location (realised through adverbials and noun phrases specifying place), characters (realised through proper nouns and pronouns) and objects (realised through nouns and pronouns). Function-advancing propositions work to develop and advance events within the text world, and are realised in verb phrases. Function-advancing propositions map on to Hallidayan processes described in systemic functional grammar [24]. Function-advancing propositions may take the form of material processes (that is, intentional, superventional or event processes), relational processes (intensive, possessive and circumstantial) and mental processes. The key benefit of Text World Theory for this project is that it provides (i) a comprehensive toolkit for accounting for all the facts that are put forward in the forensic case and (ii) an explanation of how specific elements of language trigger mental representations on the part of readers.

As for those propositions which cannot be verified as true, Text World Theory also has a way of structuring such information within the model. When linguistic indicators of hypotheticality or modality (degree of certainty) are used, the information introduced by such indicators cannot be incremented directly into the text world; instead, such information creates a modal or hypothetical world and is stored at a remove from the facts of the case. This allows for the model to differentiate between the agreed facts and possible or hypothetical situations, as specified through the linguistic choices. By using computer-based visualisations of these distinctions, we envisage that the report's readers will have a clearer understanding of the complex state of affairs therein.

IV. SYSTEM DESIGN

The aim of the advocated system is to assist linguistic using the Text Worlds Theory visualizing the structured files which is the outcome of the transformation of corpora into structured files. Figure 1 shows how linguistics visualise the TWT output file. Even though in previous section tools for document visualization were presented, here linguistics convert their documents into an XML file manually. Therefore, Figure 1 is the outcome of a manual and not automated process. It is apparent that for large files it would be very time consuming and difficult process.

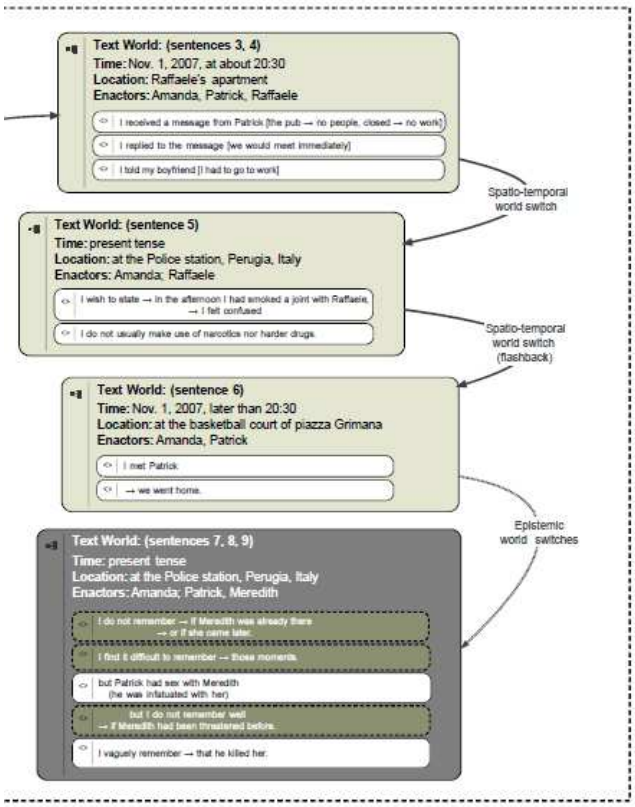

Fig. 1. Output of TWT created manually

Using TWT the statements are transformed into XML annotated files. XML is a suitable technology for transforming free text into a database. XML allows the designer to define his/hers own tags which may be organised in a hierarchical manner to structure data. Moreover, to help in structuring the data, the tags which are used in the XML contain semantic information. This built-in structure can be used both into the visualisation of the data and to process the data for more advance functionality. Since XML itself is text based, it follows that it should provide a suitable way to capture textual data. XML uses terms to describe texts that are not linked to a specific formatter and, therefore, makes documents platform-independent. [25]. An XML database facilitates complex searches, for example for loops or if conditions. A query language could automate the process of searching for data on more than one parameter within an XML document. Figure 2 shows an XML file representing a statement after the application of TWT.

In developing a method for visualising complex language data, we will use Text World Theory and XML to inform the development of a method of mark-up which we will then apply to the data, automating this as far as possible. To the knowledge of the authors there is no available system that can parse and visualize Text Worlds Theory files in a XML structures. Thus, Epicurus aims to fill this gap.

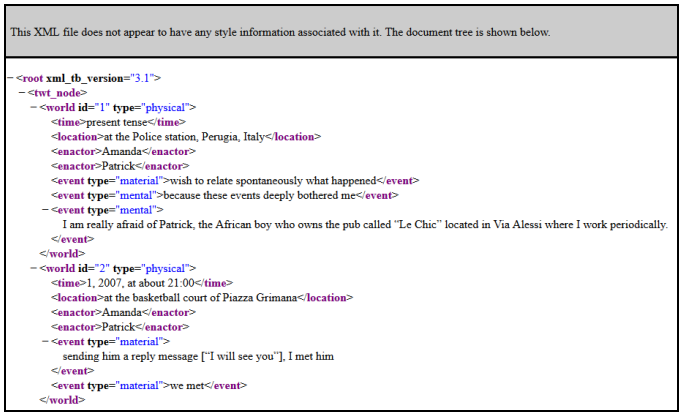

Fig. 2. *Statement transformation into XML*

V. SYSTEM DESCRIPTION

Figure 3 in blue highlighted background shows the main features of the advocated visualisation tool. The test case to verify the tool was the case of the murder of Meredith Kercher. The input for the presented tool are the statements of the Amanda Cox who was accused of co-committing the murder. The statements have been converted into annotated XML files using TWT theory rules. The features of the tool are:

- Diagrammatical form and graphical notations to present multiple type of events (nodes) in a defined space and maintaining consistency in data set representation
- Visual comparison characteristics of different nodes, node links, and time series via distinctive colour and shape schemes
- Broad overview and selected display of fine structures
- Quantifiable/statistical representations of data, i.e. Histogram and Pie Chart
- Line colour and thickness illustrate relationships or significance.

Some other functions of the tool are:

- Showing counting numbers of the following TWT features in table format. Such features are: Total number of worlds, events, locations and enactors; number of physical worlds, mental worlds, material events, mental events and number of events embedded in mental worlds.
- Showing statistical results of features using pie graphs and histograms: percentage of different of world types and different event types.
- Generating directed graphs showing the topological structure of text world where each node represents one world, using arrows to represent world switches, using different colours to represent different world types and different colours to represent different event types.
- Multiple timeline visualisation and interaction: i) generation of parallel timeline, one for each annotated statement ii) representing fully-specified and under-specified event times along the timeline. At each node information about events is provided iii) Customisable display of different type of events on the graph iv) mouse control over node position along each timeline v) Time-stamps can be shown above each node linking same events and sub-events on different timeline by dotted lines.
- Enabling editing and modifications (add, delete or change nodes) on XML inputs and auto-updating of the corresponding: tables, pie-charts and histograms, text world directed graphs and multiple timeline graphs.

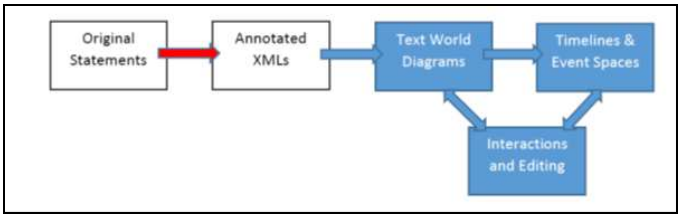

Fig. 3. *Features of Epicurus*

Epicurus reads XML files which are structured with a predefined content format. The XML file consists of a root element which in our cas2e is a statement, and the "children" of the root element are the "worlds" which take the attribute "Physical" or "Mental". Within each child information from the statement is stored such as: Enactors, Location, Time & Date and Events. Epicurus parses the XML file and represents in HTML files the information that is requested from the user.

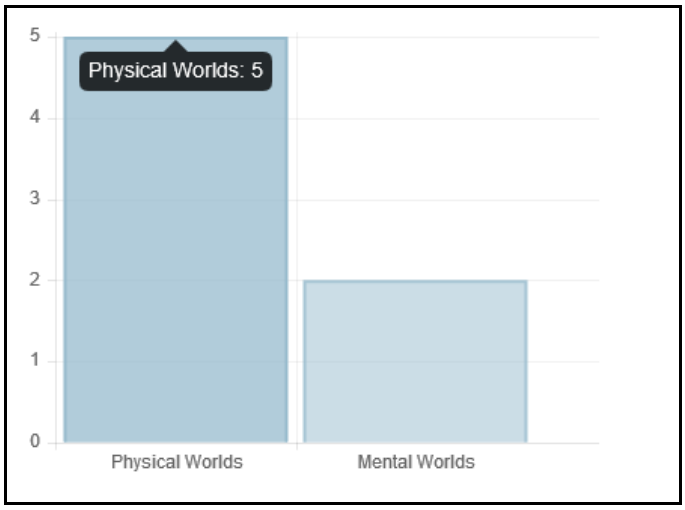

Fig. 4. Bar-chart of wolrds

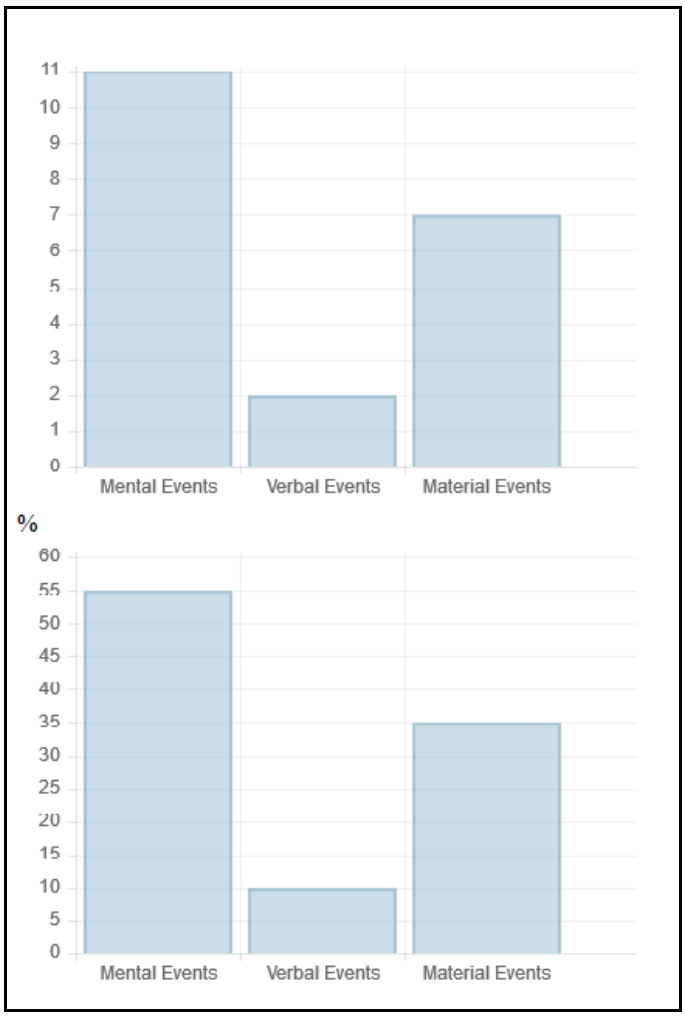

Fig. 5. *Bar-chart showing different types of events*

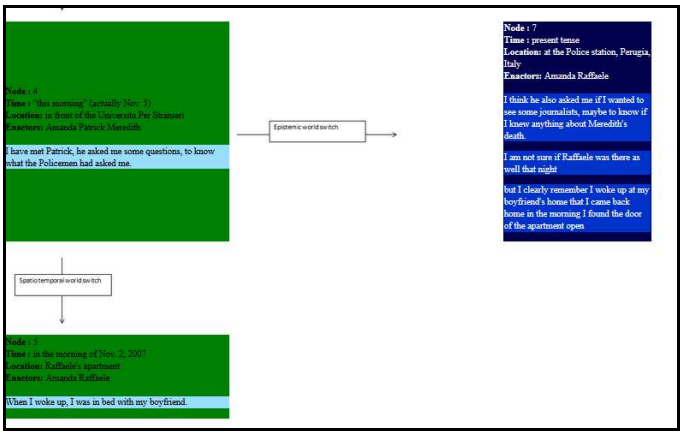

Fig. 6. Flow of worlds

Figures 4-6 show snapshots of the advocated tool. Figure 4 and Figure 5 show bar-charts which are used to show the number of different worlds and the number of different events that are contained in the worlds. Figure 6 shows the flow from one world to the next. The flow is determined from the XML file and each world is placed into a box. Different colors (for the background and the text) are used in order to make easier of the user to identify the properties of the information.

VI. CONCLUSION

A new visualisation tool for text annotation - Epicurus – has been developed during this research. It is designed to visualize forensic documents using cognitive model of human discourse processing. Epicurus aims to quantify and visualize data that have been annotated based on structured TWT framework, using XML Document Object Model (DOM) structure. The ultimate goal is to provide an accessible system with direct applications for criminal justice practitioners in their reconstructive past events and truth-rebuilding process.

REFERENCES

[1] Xu, Y., Xu, Z., Jiang, X. and Paul, S. (2010). Developing a Knowledge-based System for Complex Geometrical Product Specification (GPS) Data Manipulation. Knowledge Based Systems, 24 (1). pp. 10-22.

[2] Su, Y. and Xu, Z. (2010). Parallel Implementation of Wavelet-based Image Denoising on Programmable PC-grade Graphics Hardware. Signal Processing, 90 (8). pp. 2396-2411.

[3] Wang, J. and Xu, Z. (2013) STV-based Video Feature Processing for Action Recognition, Signal Processing. ISSN: 0165-1684. Volume 93, Issue 8. Pages 2151 – 2168.

[4] Wang, J. and Xu, Z. (2014) 'Bayesian inferential reasoning model for crime investigation', in Neves-Silva, R., Tsihrintzis, G. A., Uskov, V., Howlett, R. J., Jain, L. C. (eds) Smart Digital Futures, pp. 59-67. Amsterdam: IOS Press.

[5] Jeffries, L. and McIntyre, D. (2010) Stylistics. Cambridge: Cambridge University Press

[6] Lugea, J (2013) 'Embedded dialogue and dreams: the worlds and accessibility relations of Inception', Language and Literature 22(2): 133-53

[7] Gordon, T.F.,1993. The Pleadings Game. An Artificial Intelligence Model of Procedural Justice, dissertation.

[8] Gordon, T.F.,1995. The Pleadings Game. An Artificial Intelligence Model of Procedural Justice, Dordrecht: Kluwer Academic Publishers 1995.

[9] Hage, J., 1996. A Theory of Legal Reasoning and a Logic to Match, Artificial Intelligence and Law, Vol. 4,, pp. 199-273.

[10] Lodder, A.R., 1998. DiaLaw – on legal justification and dialog games, dissertation, Universiteit Maastricht.

[11] Prakken, H., 1993. Logical tools for modelling legal argument, doctoral thesis, Amsterdam: Vrije Universiteit.

[12] Prakken, H., 1997. Logical Tools for Modelling Legal Argument. A Study of Defeasible Reasoning in Law, Dordrecht: Kluwer Academic Publishers 1997.

[13] Verheij, B., 1996. Rules, reasons, arguments. Formal studies of argumentation and defeat, Dissertation, Universiteit Maastricht.

[14] Verheij, Bart, Hage, Jaap, Lodder, Arno R., 1997. Logical tools for legal argument: A practical assessment in the domain of tort. Proceedings of the International Conference on Artificial Intelligence and Law, pp. 243-249.

[15] Verheij, B., 1998 Dialectical Argumentation with Argumentation Schemes: An Approach to Legal Logic. Artificial Intelligence and Law, 11, 167-195.

[16] Verheij, B., 1998. ARGUE!—An implemented system for computer-mediated defeasible argumentation, in: H. La Poutré, J. van den Herik (Eds.), NAIC'98. Proceedings of the Tenth Netherlands/Belgium Conference on Artificial Intelligence, CWI, Amsterdam.

[17] Verheij, B., 1998. ARGUMED—A template-based argument mediation system for lawyers, in: J.C. Hage, T.J.M. Bench-Capon, A.W. Koers, C.N.J. de Vey Mestdagh, C.A.F.M. Grütters (Eds.), Legal Knowledge Based Systems. JURIX: The Eleventh Conference, Gerard Noodt Instituut, Nijmegen.

[18] Reed, C.A., and Rowe, G.W.A, 2004. Araucaria: Software for argument analysis, diagramming and representation. International Journal of Artificial Intelligence Tools,14, 961-980.

[19] Monroe, M., Lan, R., Lee, H., Plaisant, C., and Shneiderman, B., 2013. Temporal event sequence simplification. Visualization and Computer Graphics, IEEE Transactions

[20] Plaisant, Catherine, Milash, Brett, Rose, Anne, Widoff, Seth, Shneiderman, Ben,1996. LifeLines: visualizing personal histories. Conference on Human Factors in Computing Systems - Proceedings, pp. 221-227

[21] Murphy, S., Mendis, M., Hackett, K., Kuttan, R., Pan, W., Phillips, L., Gainer, V., Berkowicz, D., Glaser, J., Kohane, I., Chueh, H.,2007. Architecture of the open-source clinical research chart from informatics for integrating biology and the bedside. Proc. AMIA.

[22] Wongsuphasawat, Krist, John Gomez, Catherine Plaisant, Taowei Wang, Ben Shneiderman, and Meirav Taieb-Maimon. "LifeFlow: Visualizing an Overview of Event Sequences." CHI-2011, n.d., 1747–56.

[23] Werth, P. (1999) Text Worlds: Representing Conceptual Space in Discourse. London: Longman.

[24] Halliday M.A.K. 1970. A Course in Spoken English: Intonation . Oxford: Oxford University Press.

[25] Kroeze, J.H, Bothma, T.J.D., and Matthe, M.C.,2010. Constructing an XML database of linguistics data. VTC School of Information Technology Collections (99).