Text Analyzing Tool

Aditi Mallavarapu
University of Illinois at Chicago
Chicago, USA
amalla5@uic.edu

ABSTRACT

Language hides some inherent relations among its constituents. Illustrating their relations to understand the relations though may seem trivial but has been considered to be very challenging. As NLP problems are being more complex and the amount of data is increasing rapidly, NLP softwares are expected to be robust and comprehensive. Aiding the analysts with proper tools to visualize and to analyze these complex relations is necessary. Having a good analysis tool would also help NLP expand to other technical domains such as biology, chemistry and many others. The tool that we have developed helps in analyzing the relationship among words and between words and other components of a sentence like phrases, or sentences etc. in the text by annotating the relations between these components.

Author Keywords

Text annotations; graph visualizations; Natural language processing.

INTRODUCTION

Annotation can often be time consuming and very domain specific. Although there are tools that would reduce the annotation time and cost, but certain relations get masked. Our effort was to design a tool that not only makes the annotation faster but makes the analyzation after annotation more intuitive. We intend to elicit all the relationships in the text, not only the primary ones and be able to differentiate among them as per the domain specific needs. Our idea was to design two views to make the tool easier to use. The text-view which would have the text and the relationships with arrows spanning across the words. The second view is the staircase view which would represent each element like the paragraph, and sentence as a series of word staircase and relationships with arrows from one stair to the other. At this time we have only designed the staircase view. We have used different texture and color coding to differentiate between the types of relationships and the relationships which span outside the sentence, to make it more intuitive. The staircase view also allows to zoom into different levels of the text. These two views can be used in conjunction to view and analyze the relationships for the text.

RELATED WORKS

Two existing tools that we know of are BRAT (and its predecessor STAV) and Odin Open Domain Rule Visualizer. Both tools are fairly new for public use (BRAT was published in 2012 and Odin was 2015), therefore they are not yet optimized for NLP researchers. Although BRAT comes with a plenty of functionalities, it only supports node-to-node (indicating connection between words) connections. We plan to support node-to-edge (relationship between words and other relationships), edge-to-node (relationship between other relations and words), and edge-to-edge (relationships between relations) connections as well. Another problem with BRAT is to identify the end connections. It is cumbersome to see the other end of the connection when the connection spans across many lines.

Although ODIN was able to address this issue the user has to encode the relationships in a rule based language, which might not be very friendly to scientists who deal with only domain specific stuff. Drawing on this, there is a need for a tool that would be able to elicit the relationships and also make the tool user friendly for the domain experts and domain independent to be specific.

ANALYSIS TASKS

We are using two JSON files:

edge.json which gives us the specifications about a certain edge.

Eg:
"Id":"E1","sourceid":"T44","destinationid":"T9","label":"Theme" ,"type": “n2n”

ID represents the ID of a specific relationship (an unique edge). Considering the example above, it describes an edge between two words. Each edge is described by the above fields. Sourceid is the source word or source edge from where the relationship starts and destinationid is the ending word or ending edge where the relationship ends. Label is the name of the relationship through which both are connected. Type describes the type of the edge to be one of the following:

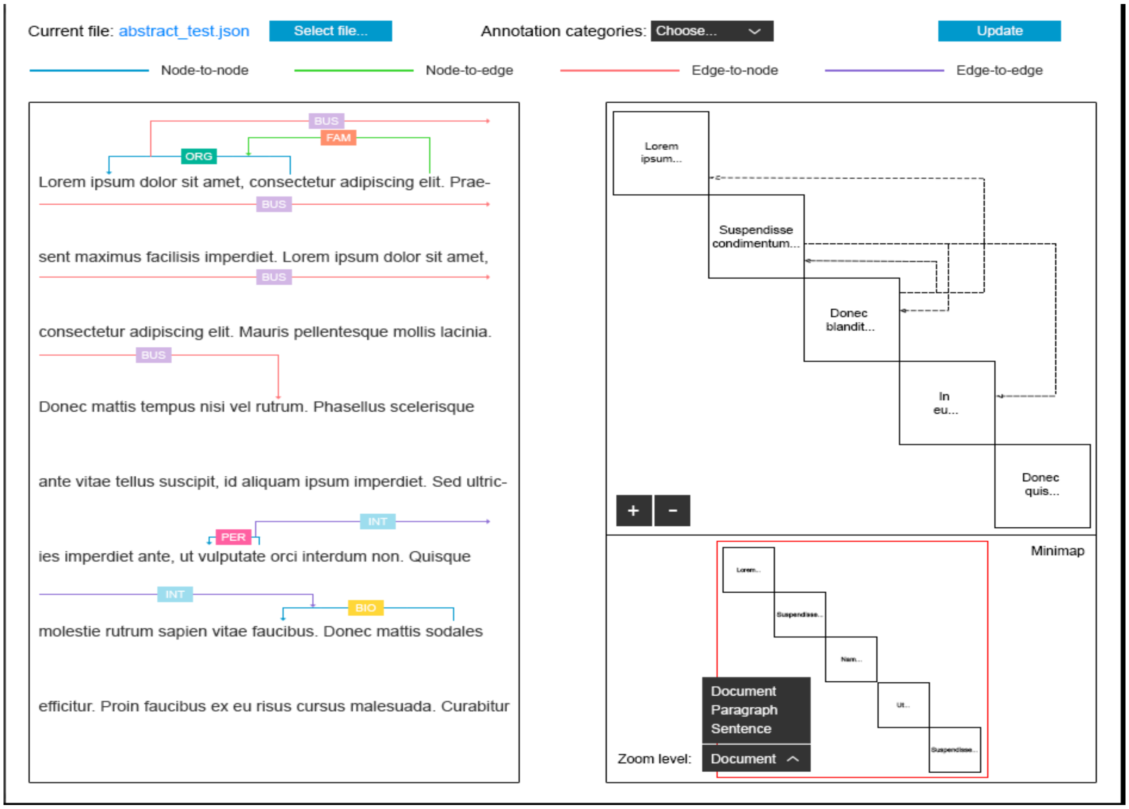
N2N : Node to node (edge between two words)

N2E: Node to edge (edge between a word and another edge)

E2N: Edge to node (edge between another edge and a word)

E2E: Edge to edge (edge between two edges)

Each word in the document is given an id based on its position in the actual text. Hence in the above example, the source word is at T44 (44th word of the document) is connected to T9 (9th word of the document) through the relationship “Theme” and the ID of this relationship is given by E1. The type is used to filter out certain edges for ease. This is the case of word to word relationship.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Prae-
sent maximus facilisis imperdiet. Lorem ipsum dolor sit amet,
consectetur adipiscing elit. Mauris pellentesque mollis lacinia.
Donec mattis tempus nisi vel rutrum. Phasellus scelerisque
ante vitae tellus suscipit, id aliquam ipsum imperdiet. Sed ultric-
ies imperdiet ante, ut vulputate orci interdum non. Quisque
molestie rutrum sapien vitae faucibus. Donec mattis sodales
efficitur. Proin faucibus ex eu risus cursus malesuada. Curabitur

Figure 1: Proposed Conceptual Design: the text view (left) and the Staircase view (right)

"Id":"E9", "sourceid":"T45", "destinationid":"E8", "label":"Cause",”type”: “e2e”

Similarly in the above example, the source word is T45(45th word in the document) which is connected to the to E8(the 8th edge in the document as per the file edge.json) through the relationship tagged as Theme and it’s given an ID of E1. This is the case of word to edge relationship.

2) labelleddata.json which given the identification of every word in the document

Eg: "Id": "T1","wordindex": 1,"word": "acid"

The position of the word in the document is given by the ‘wordindex’ and the word is given the tag ‘word’ and the Id of the word is given by ‘Id’. In the above example, the position of the word ‘acid’ in the document is 1 and the id of the word is T1. We intend to elicit for future works if the word carried a specific tag (belonged to a specific group) if the need be.

CASE STUDY

We designed in the initial pass the staircase view with multiple zoom levels. The edges are color coded to represent different source and destinations respectively. The node to node relationships are represented as blue, the node to edge relationships are represented by green, the edge to node represented by pink and edge to edge relationships represented by purple. The webpage has “Reset Viz” button, it resets the visualization to the paragraph zoom level and resets the checkboxes. The user can use the checkboxes to select specific types of edges in order to analyze the relationships.

The webpage initially opens up in the paragraph level where the edges show the relationships amongst the words inside paragraphs. The user can analyze the relationships at the paragraph level. Each edge can be analyzed by hovering over the edges, on hover, the relationships are displayed with the each edge enclosed brackets with the label between the source and destinations. The edge description can be viewed at any zoom level.

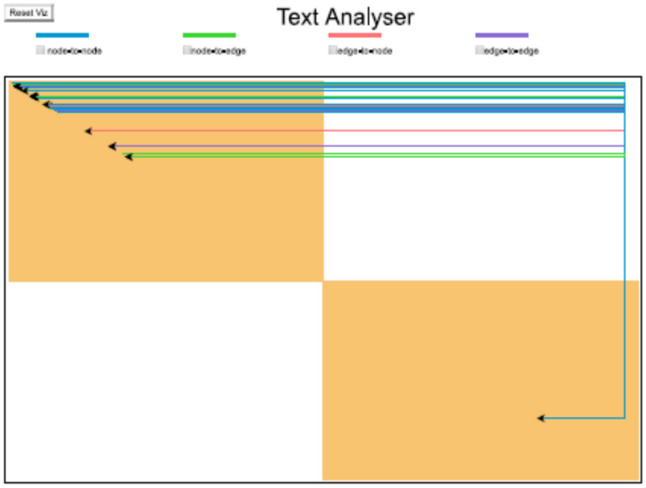


Figure 2: Staircase view: Paragraph Level

This particular data has two paragraphs represented by two rectangle staircase in Figure 2.

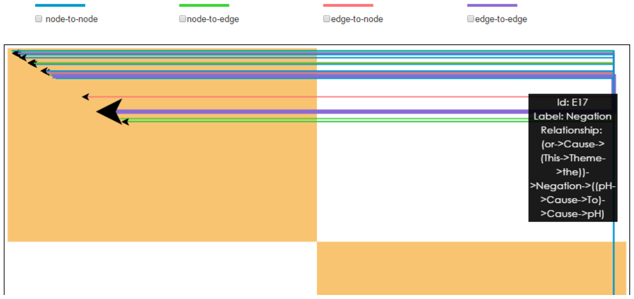


Figure 3: The edge description with edge id label and the relationship

The user can click the paragraph to get to the second zoom level. The second zoom level is the sentence level, each sentence in the paragraph (that has been clicked) is represented as a rectangle and the whole paragraph as a series of sentence staircase. The edges here only portray the relationships between the words and edges in those sentences, if one of the relationship end is in another sentence not currently being displayed on the screen it is displayed as a dotted edge to the corner of the screen. Figure 4 depicts the sentences in the first paragraph , which contains 8 sentences displayed as a staircase.

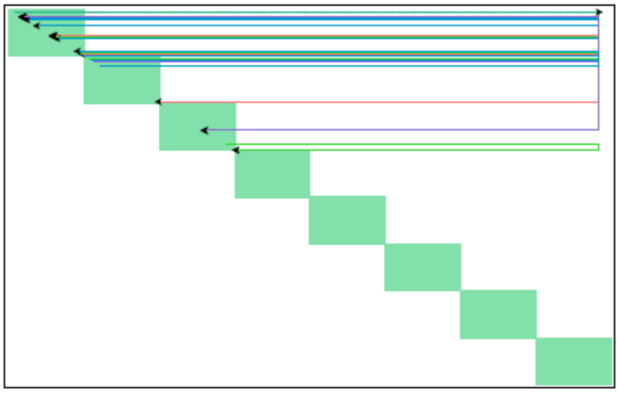


Figure 4: Staircase view: Sentence level

The user can click a particular sentence to see the relationships between the words in that sentence. This is the third zoom level the word zoom level. The dotted edges would mean the other end of the edge is in another paragraph or another sentence of the same paragraph. Figure 5 shows the 48 words in the first sentence of the first paragraph as a word staircase.

These zoom levels enable the analyst to focus on the part of the text that one wishes to analyze, without losing the order of the words and also maintaining the context within the paragraph and the sentences. We wish to integrate this visualization with the conceptual textual view like ODIN and BRAT currently use. The use of both the view may add more context to the analysis and make the analysis more intuitive.

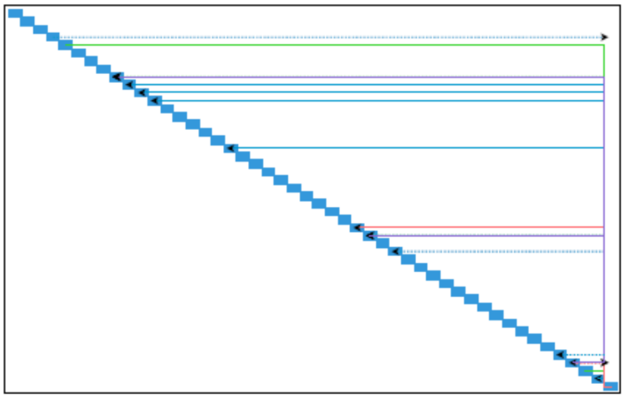


Figure 5: Staircase view: Word level

EXPERT FEEDBACK

The experts were thoroughly impressed by the design especially the stair-case diagram. They really liked the idea which they said, like they haven't seen that in context of text annotation.

They expect cross sentence and cross paragraph relationship in the future, so that will be pretty challenging with the recent architecture.

CONCLUSIONS AND FUTURE WORK

We wish to automate this work by enabling the user to choose the text file and elaborate on the relationships in simple English language independent of the domain. The application would take these files and internally convert them into a format suitable for the tool to visualize any text. Further we would want to enable the user to annotate new relationships by simply using the either the staircase view or the text - view.

Also we could extend the idea of the relationships from single words to a group of words which would explain phrases being in related.

ACKNOWLEDGMENTS

Sample text: We thank all the volunteers, and all publications support and staff, who wrote and provided helpful comments on previous versions of this document. Authors 1, 2, and 3 gratefully acknowledge the grant from NSF (#1234-2012-ABC). This is just an example.

REFERENCES

1. Pontus Stenetorp et al. *BRAT: a Web-based Tool for NLP-Assisted Text Annotation*. 2012. EACL '12 Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics.

2. Marco A. Valenzuela-Escárcega, Gus Hahn-Powell, Thomas Hicks, Mihai Surdeanu. *A Domain-independent Rule-based Framework for Event Extraction*. 2015. ACL-IJCNLP 2015.

3. Marco A. Valenzuela-Escárcega, Gus Hahn-Powell, Thomas Hicks, Mihai Surdeanu. *Description of the Odin Event Extraction Framework and Rule Language*. 2015. Retrieved from https://arxiv.org/abs/1509.07513.