Visualization for Coreference Resolution Error Analysis

Colin Mayer and Francois Chaubard
CS 448B Data Visualization
Stanford University
(cmayer,fchaubar)@stanford.edu

ABSTRACT

Graphically encoding information in text is inherently difficult and consequently the field of Natural Language Processing (NLP) suffers greatly from a lack of effective data visualizations. This problem is especially salient when designing NLP systems, as the output is rarely intuitive and error interpretation can be unecessarily time consuming. One NLP task that suffers in particular is coreference resolution. Coreference resolution is a cluster based analysis and the permutations of possible errors make visualizing the results challenging. This paper proposes a new approach to visualizing the results from a coreference resolution system. The approach focuses on the needs of the system designer and leverages a combination of drill-down and brushing and linking visualization strategies to provide an intuitive interface for extracting actionable error information.

INTRODUCTION

Coreference Resolution is the task of computationally grouping references to the same physical entity (i.e. person, place or thing). An individual reference is called a "mention", and its group a "cluster". This task is often accomplished by constructing a model that compares mentions pairwise, then clusters those that refer to the same entity. The output of the model is clusters of mentions such that every mention in a given cluster is homogenous, that is they all refer to the same entity. A major challenge in coreference resolution is evaluating incorrect solutions. We call the clustering solution outputted by the model, the "guess" clusters, and compare our results to the correct "gold" clusters. Quantifying the amount of error in a "guess" cluster is non-trivial, since there can be any combination of overlap and underlap with other "gold" clusters. Scoring systems exist that attempt to solve this problem, but they are not very useful for system design. Most improvements in coreference system design come from evaluating individual documents and errors. Thus, error analysis in coreference resolution is extremely important. In this paper we explore a platform for improved error visualization of coreference system output.

Text Visualization Challenges

When creating data visualizations, the first step is to evaluate your data types. Commonly data is broken down into three categories: nominal, ordered and quantitative. Nominal, or categorical, data consists of discrete data that falls into separate categories with no concept of ordering. Ordered data, as the name implies, is discrete data that has a sense of relative order but not necessarily global value. Quantitative data, perhaps the most familiar, is continuous data with a global value. Most modern data visualizations follow the models for representing information in graphics presented in [2]. Bertin proposes a hierarchy of "Levels of organization" shown in Figure 1, where the most informative methods for encoding information in graphics are (in order): position, size, value, texture, color, orientation and shape. However, most of these encodings are note effective for visualizing data in text documents. The position, shape and value of text cannot be altered without changing its meaning, and therefore cannot be used. While size can be encoded in text, a good example is word clouds, the inherent variability of word length makes it a poor choice. Longer words will appear larger than shorter words with the same font size even though they encode the same data value. Orientation disrupts the readability of text, particularly in the context of a document, and likewise is a poor choice. This leaves only two effective encodings for data in text documents: color and texture. Figure 1 shows that color can only effectively encode nominal data, while texture can encode nominal data, and to a lesser extent ordered data. With such a limited set of encodings, creating effective data visualizations for text documents is particularly challenging. A common solution, and the one used in this paper, is to use interaction to allow the viewer to control what information is being shown in any given view.

Coreference Resolution Evaluation

For most machine learning problems, constructing the error function is pretty easy (i.e. least squares, SVM, Kmeans, Softmax). Furthermore, the Precision/Recall (or confusion matrix for class size greater than 2) is easy to compute, visualize and comprehend. This makes correcting for mistakes in your model straightforward. In coreference resolution, this process is a lot more difficult. Not only does the sys-

Position	N	O	Q
Size	N	O	Q
Value	N	O	Q
Texture	N	O	
Color	N		
Orientation	N		
Shape	N		

N Nominal
O Ordered
Q Quantitative

Figure 1. Bertin's "Level's of Organization"

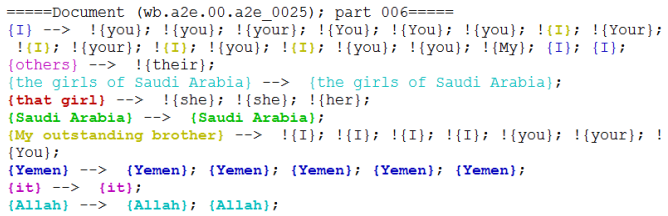

Figure 2. Previous Error Visualization

tem need to create homogeneous clusters for all mentions but it has to "guess" the correct number of unique entities in the document to begin with. One issue is the lack of a single error function to optimize for. We saw in our experiments with singleton and complete clusters, that the traditional NLP measure of accuracy and recall cannot accurately describe system performance. There are a number of scores that try to capture this objective properly (B3, MUC etc) [5][1] but an effective evaluation still requires looking at a combination of their various parameters (i.e B3 F1, B3 Recall, B3 Precision, MUC F1,). Another issue is determining the causes for model errors. Here we note a Confusion Matrix would have negligible value due to its size and sparseness. Coreference resolution is a unique error analysis task that not only requires a bespoke error formula, but also a specialized error visualization that allows the user to understand what mentions the model is getting wrong and why.

PREVIOUS VISUALIZATION

Figure 2 shows an example of the current error output for a coreference system. The output shows only the mention clusters and uses color to encode the different gold clusters. If a guess mention is incorrectly clustered the output annotates the mention using an exclamation point and possible change in color. The output does a good job of presenting entities and their corresponding clusters to the user, however error details are not intuitively represented and valuable contextual information is lost. The re-use of color encoding for representing incorrectly clustered mentions is particularly confusing without a key. Additionally mention clusters are presented in isolation, and finding context requires opening the document in a separate view and comparing. Since mentions often consist of identical strings there can be ambiguity in the mapping of the output mentions to the words in the document. Context is an integral component of coreference resolution system design, particularly for analyzing pronoun coreference. For example, the Hobb's algorithm for pronoun coreference resolution operates entirely on the context of the pronoun. Consequently, a time consuming and ambiguous process for resolving mention context is a great hindrance to coreference system development.

DESIGN APPROACH AND RATIONALE

Our visualization builds on the strengths of the previous representation, but leverages interaction to present the data in context with the same level of detail. The visualization breaks down the error analysis into two distinct views. The initial view presents the entire document along with gold clusters and basic error information. Each mention is bolded and colored to distinguish it from normal text. As in the previous error representation, each gold cluster is represented by a unique color. Here color represents a nominal encoding and the scheme was generated by a cartography tool [3] for maximum differentiability. To find incorrect mentions, we use a greedy approach similar to the B3 score calculation to pair gold and guess clusters by maximum overlap. We then find any mention that is incorrectly added to or ommitted from a gold cluster and flag it as incorrect. Incorrect mentions are marked with an underline. While this approach is far from complete, it provides a straightforward entry point for user interaction. In the document view the overall structure of the document and the context of the gold clusters is easily visible. Additionally, incorrectly classified mentions are underlined in their exact positions providing valuable insight into the relationship between system errors and mention context. A key is shown in the lower corner of the visualization so the various encodings for both the document and detailed view are unambiguous.

The detailed view is activated by clicking on a mention. In the detailed view both the gold and guess clusters that include the selected mention are brought to the forefront, while all other text is dimmed and grayed out. The gold cluster for the selected mention maintains the same color and is redundantly encoded with a shadow texture. This further differentiates gold cluster mentions from any mentions that are incorrectly added to the guess cluster. Mentions in the guess cluster are boxed. This view clearly presents the intersection and deviation of the guess and gold clusters. "Correct" mentions in the cluster overlap are easily seen as shadowed text of the cluster color that is boxed. Mentions that are incorrectly added to the guess cluster are distinguishable as being boxed but having a different color and no shadow. Finally mentions that are incorrectly ommitted from the guess cluster are seen as shadowed text of the cluster color without a box. While the clusters of interest are brought to the forefront in this view, mention positions are unchanged and the document text is only faded to maintain the overall sense of context.

This interaction approach represents a combination of two popular visualization techniques known as drill-down and brushing and linking. The drill-down technique facilitates detail extraction and happens when a user selects a specific mention to get the details of its gold and guess clusters. Brushing and linking occurs in this view when a mention is selected or "brushed" and other mentions in the same clusters are consequently "linked" with shadows and boxes. Clicking away from a mention, or on the currently selected mention disables the detailed view and returns the user to the initial view.

IMPLEMENTATION

The error analysis process for coreference resolution needs to be fast and intuitive, since the process will be repeated many times. Our approach was to provide a solution that could be run locally, and hooked into an existing coreference system. We wrote the visualization as a JavaScript web application requiring only a modern web-browser. The coref-

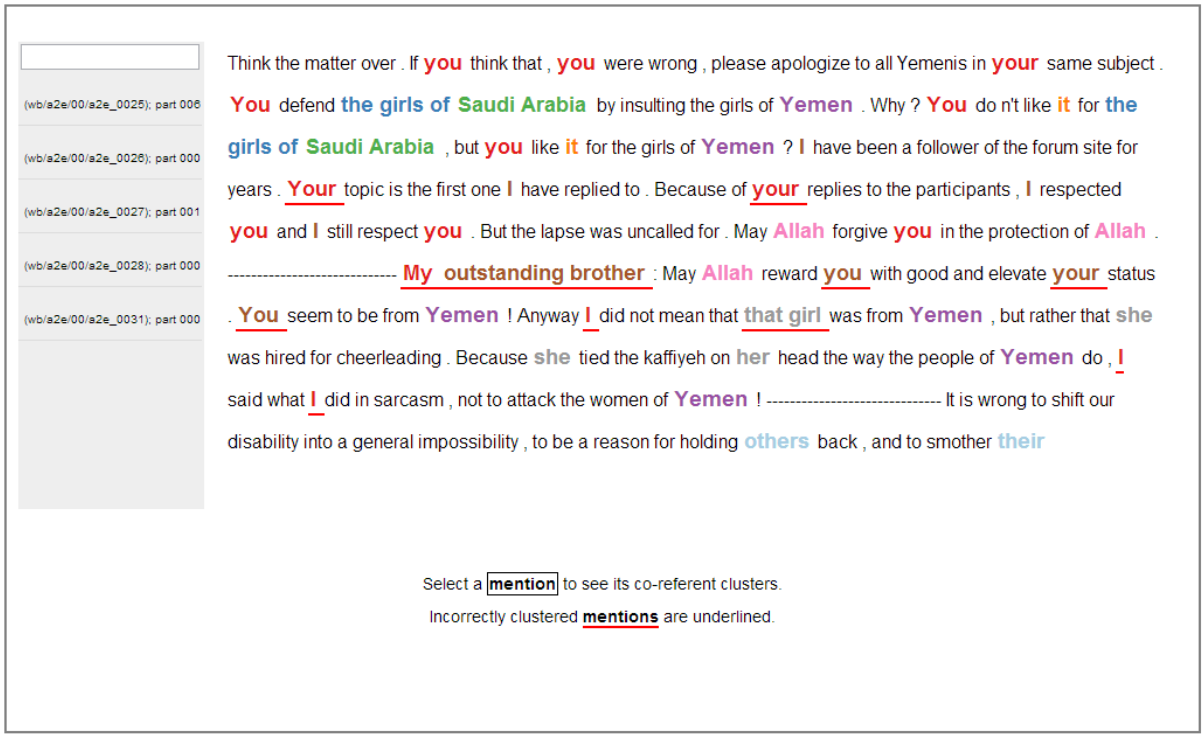

Figure 3. Initial Document View

Figure 4. Detailed Cluster View

erence system hook outputs a properly formatted JSON file that can be dragged into the web application creating the visualization. We noticed that we were often searching for a specific mention that our system was getting wrong, so we implemented a custom search method that searches through all the documents in both content and title to minimize document discovery time.

RESULTS

An example error output is shown in Figures 3 and 4. In the document view you can see all the gold clusters marked in their respective colors, and the errors in the guess clusters underlined in red. At first glance it is clear that most errors occurred when trying to resolve the pronouns "you" and "I" as the speaker changes in the text. More information on the guess clusters that created these errors can be seen in the detailed view.

Figure 3 shows the detailed view that is activated after a user clicks on one of the early "you" mentions. The corresponding gold cluster that references the second speaker is highlighted in red with a red shadow, while the guess cluster is shown with a black border. It is easy to see that the guess cluster omits a few "you" mentions in the first section, then completely fails to recognize the switch of speaker adding an incorrect "you" mention and omitting all the "I" mentions in the second section. Since the previous visualization doesn't show context, recognizing that this error occurred because the system failed to recognize a change in speaker would require multiple passes of the document cluster list. In this visualization the behavior is immediately apparent after one click to drill-down to the clusters of interest.

Future Work

Future work will explore better integration with coreference system codebases. In its current implementation the system requires a user to output a JSON file form the coreference system, then find it on the file system and drag into a web browser. A more seamless integration that automated this process and displayed the error visualization immediately upon completion would be ideal.

Using similar drill-down and brushing approaches could be used to develop visualizations for different NLP tasks as well. For example, when doing syntatic or semantic parsing, the gold parse trees could be shown with incorrectly parsed nodes or subtrees highlighted. Clicking on an incorrect node could then expand that node into the guess and gold subtrees so the user can easily see where the errors occurred. For a machine translation task, the traditional alignment matrix could be replaced with a bundled edge graph to show gold alignments. Bundled edge graphs group edges that are in close proximity and follow similar paths. They are very useful for pattern detection of block re-alignment within or in between data sets. Once again incorrectly aligned words would be flagged, and a drill-down with brushing and linking used to highlight the gold and guess alignments for the word when selected.

REFERENCES

1. A. Bagga and B. Baldwin. Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–6. Citeseer, 1998.
2. J. Bertin. Semiology of graphics: diagrams, networks, maps. 1983.
3. C. Brewer and M. Harrower. Colorbrewer 2.0 color advice for cartography, 2013.
4. K. Raghunathan, H. Lee, S. Rangarajan, N. Chambers, M. Surdeanu, D. Jurafsky, and C. Manning. A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 492–501. Association for Computational Linguistics, 2010.
5. M. Vilain, J. Burger, J. Aberdeen, D. Connolly, and L. Hirschman. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th Conference on Message Understanding*, MUC6 '95, pages 45–52, Stroudsburg, PA, USA, 1995. Association for Computational Linguistics.