AmbiCoRefVis: A Tool for Visualizing Coreferential Ambiguity

Patrick Paetzold*, **Lukas Beiske***, **Mark-Matthias Zymla***, **Massimo Poesio**†, **Miriam Butt***, **Daniel Weiskopf**‡, **Oliver Deussen***
*University of Konstanz, Germany
{patrick.paetzold, lukas.beiske, mark-matthias.zymla, miriam.butt, oliver.deussen}@uni-konstanz.de
†Queen Mary University of London, United Kingdom and University of Utrecht, The Netherlands
m.poesio@uu.nl
‡University of Stuttgart, Germany
daniel.weiskopf@visus.uni-stuttgart.de

Abstract

Situations of ambiguity and uncertainty in the annotation of discourse interpretation tasks, such as anaphoric reference, are common, but existing annotation tools typically only support visualization at the local level (i.e., visualizing more than one mention of a possible antecedent) rather than globally (i.e., visualizing multiple coreference chains), as the latter is a complex problem. In this paper, we introduce the interactive visual analysis tool *AmbiCoRefVis*, developed to display multiple global interpretations of a referring expression. We evaluate it with the *Phrase Detectives* corpus.

Keywords: anaphoric reference, ambiguity, coreference, uncertainty, visualization of linguistic information

1. Introduction

Anaphoric reference (or coreference) interpretation recognizes if a referring expression mentions an entity already introduced in discourse, as in (1), where the first mention of *Alice* introduces a new discourse entity that is then anaphorically referred to in the text, e.g., via *her*.

(1) [Alice]$_i$ was beginning to get very tired of sitting by [her]$_i$ sister on the bank, and of having nothing to do: once or twice [she]$_i$ had peeped into [the book]$_j$ [her]$_i$ sister was reading, but [it]j had no pictures or conversations in [it]$_j$, 'and what is the use of a book,' thought [Alice]$_i$ 'without pictures or conversation?'

As (1) shows, the most important entities in a text (Alice and the book in this example) are typically mentioned several times. The chain of references to the same entity is called a coreference chain. Modern tools for annotating and visualizing anaphoric information, such as MMAX2 (Müller and Strube, 2006), support the visualization of such coreference chains, as illustrated in Figure 1.

However, anaphoric expressions cannot always be resolved in context (Poesio and Reyle, 2001; Poesio et al., 2006; Versley, 2008; Recasens et al., 2011), a situation of uncertainty that occurs in problems of linguistic annotation more generally (Beck et al., 2020). For example, the pronoun *it* in the last utterance in (2) from the TRAINS domain of the ARRAU corpus (Poesio et al., 2024) could refer to *engine E2*, *the boxcar that's at Elmira*, or even both. Human subjects do not always agree on the antecedent in such examples either (Poesio et al., 2006; Versley, 2008).

(2) **M:** [...] why don't we couple [engine E2] to [the boxcar that's at [Elmira]]
S: okay
M: and um ... send [it] to Corning

Such cases raise a major issue for conventional methods for visualizing anaphoric reference, as multiple coreference chains–potentially combinatorially many–need to be computed and displayed.

In this paper, we introduce the novel interactive visual analysis tool AmbiCoRefVis,[1] designed to support linguists in exploring anaphoric references even in the presence of such commonly occurring ambiguities (Poesio et al., 2019). We demonstrate the tool with respect to the *Phrase Detectives* corpus (Yu et al., 2022). This corpus is ideal for our purposes as it contains anaphoric annotations by thousands of subjects collected through the *Phrase Detectives* game-with-a-purpose (GWAP).

2. Background and Related Work

In this section, we provide an overview of the sources of ambiguity within anaphoric interpretation corpora, and provide background on the visualization of ambiguity and uncertainty.

2.1. Anaphoric Reference Terminology

We call an annotated span of text in a corpus a *markable* (or *mention*). Any individual, event, or abstract object referred to in a text (typically via a noun phrase) is called a *(discourse) entity*. An *anaphoric expression* establishes a link to a previously mentioned entity, its *antecedent*. All *nominal*

[1] https://graphics.uni-konstanz.de/AmbiCoRefVis/

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 1842–1855
11–16 May 2026. ©ELRA Language Resources Association (ELRA), 2026

entities (NE) referring to the same discourse entity form *coreference chains*. When a markable introduces a new entity to the discourse, it is called *discourse-new* (DN), and *discourse-old* (DO) when referring back to one (Prince, 1992).

2.2. Tools for Annotating and Visualizing Anaphoric Reference

The annotation and analysis of anaphoric information are supported by a variety of different purpose-built tools and frameworks. Early examples include tools for laptop or desktop use such as CALLISTO (Day et al., 2004) and PaLinkA (Orăsan, 2003), as well as MMAX (Müller and Strube, 2006), possibly the most widely used among these tools and our main comparison here. All of these tools provide an easy-to-use visual interface for annotators which highlights annotations in the text and displays coreference chains in a visual way.

To collect a large number of annotations, web-based tools were developed. These include the pioneering SERENGETI (Stührenberg et al., 2007), CoRefi (Bornstein et al., 2020), which provides an easy-to-use crowd-sourced interface but does not offer any visualization of the results. WebAnno (Eckart de Castilho et al., 2016), a comprehensive web-based platform, supports creating and analyzing syntactic and semantic annotations generated by potentially multiple annotators. In addition, it provides functionality for computing inter-annotator agreement. Glozz (Widlöcher and Mathet, 2012) also offers multiple functionalities, such as an in-situ text overlay and a more abstracted graph view. SACR (Oberle, 2018) is an annotation tool designed with a strong emphasis on usability, offering an annotator-friendly interface that streamlines the otherwise highly manual task of coreference annotation. It provides a visualization of coreference chains in a tree-like structure.

2.3. Anaphoric Ambiguity in Corpora

The findings about human disagreement on anaphoric interpretation (Poesio and Reyle, 2001; Poesio and Artstein, 2005; Poesio et al., 2006; Versley, 2008; Recasens et al., 2011) led to the creation of anaphorically annotated corpora in which ambiguity is preserved (Krasavina and Chiarcos, 2007; Versley, 2008; Recasens et al., 2012; Ogrodniczuk et al., 2013; Uryupina et al., 2020; Poesio et al., 2024; Ye et al., 2023). In such corpora, ambiguity and disagreement are treated as information, rather than as noise to be filtered out or as an intermediate stage of interpretation that awaits resolution.

For this paper, we use the latest release of the *Phrase Detectives* corpus (Yu et al., 2022), which, to our knowledge, is the largest anaphorically annotated corpus preserving multiple interpretations for each anaphoric expression. In *Phrase Detectives*, ambiguity was identified *implicitly* by receiving judgments from participants (25 on average) and then identifying possible interpretations through probabilistic aggregation methods (Poesio et al., 2019). Our tool AmbiCoRefVis was designed to use the MAS-XML format developed for *Phrase Detectives* (Poesio et al., 2012) as the primary input format.

While the scheme used in *Phrase Detectives* for anaphoric annotation and for ambiguity is simple, the judgments required from players are not simply to decide whether an expression is DN or DO. First, players have to decide whether a markable is *referring* or *non-referring*. Then, if it is referring, they have to decide whether it is DN, e.g., *Mary* and *John* in (3), or DO, e.g., *they* in (3). Players may also identify split-antecedents, i.e., *Mary* and *John* in (3), or non-referring markables such as expletives and predicative NPs. In this paper, we concentrate on the ambiguity between DN and DO, and on different antecedents for a DO anaphoric expression.

(3) [Mary]$_i$ met [John]$_j$, and [they]$_{i/j}$ talked.

Tools such as MMAX2 are commonly used for anaphoric reference annotation and visualization, as they support the definition and visualization of coreference chains (Müller and Strube, 2006), as illustrated in Figure 1, in which the coreference chain of references to the entity Biles is visualized. However, such tools have not been designed to facilitate a detailed analysis and visualization of ambiguity in coreference chains.

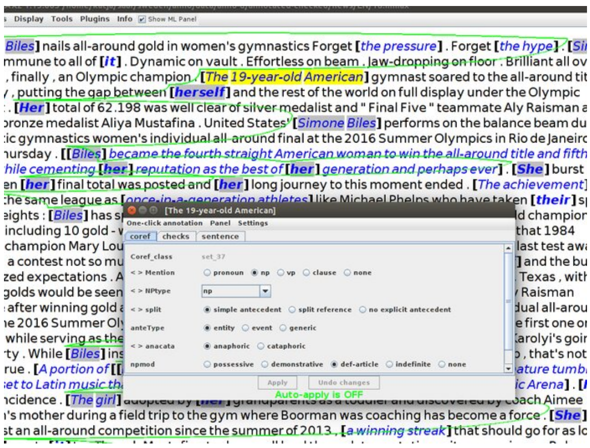


Figure 1: Coreference chains in MMAX2

This interface can be used to display two alternative interpretations for one markable at a time. For example, in the ARRAU corpus, alternatives can be displayed by marking anaphoric relations using *pointers* instead of *sets* (Müller and Strube, 2006; Uryupina et al., 2020; Poesio et al., 2024). However, these links are local and are not embedded in full coreference or ambiguity chains. Such chains would have to be backtraced via various IDs. Thus, it is difficult to use such annotations to analyze ambiguity as it unfolds over the discourse and to understand how it might interact with other entities referred to in the discourse. The tool introduced in this paper addresses this need by visualizing ambiguity as well as interannotator disagreement directly in the annotated text.

2.4. Uncertainty Visualization

Ambiguity in language as well as interannotator disagreements both represent situations of *uncertainty*. We therefore build on and extend existing research on uncertainty visualization in the development of our tool.

Uncertainty visualization is a long-standing problem and major research challenge of visualization research (Johnson, 2004), and numerous visualization techniques are available, see, for example, Pang et al. (1997), Bonneau et al. (2014), Padilla et al. (2021), or Weiskopf (2022). Importantly, general recommendations have been developed on how to appropriately represent uncertainty visually (MacEachren et al., 2012). To date, most of the work on uncertainty visualization addresses applications within the natural or engineering sciences, such as visualizing simulations or measured data.

Less work is available on abstract data, such as graphs (networks). Graphs are relevant for this paper because coreferential information can be modeled as a graph. In Schulz et al. (2017), the authors encode uncertainty by altering the graph's layout. An overview of graph visualization without uncertainty is provided, e.g., by Battista et al. (1998) and Beck et al. (2017). However, there is little work on graph visualization with uncertainty, let alone the specific problems of graphs related to linguistic data. Haghighatkhah et al. (2022) identify various sources of uncertainty in the visual analysis pipeline, focusing on textual data. Among the sources of uncertainty is comprehension uncertainty. This is similar to both of our problems of ambiguity and interannotator disagreement. Finally, to build trust in a visualization system, it is key to communicate the source of the uncertainty (Sacha et al., 2016) as part of the visualization.

3. Design Methodology

Before presenting the individual components of AmbiCoRefVis, we briefly outline its conceptualization and the underlying design process. The system emerged from a collaboration between linguistics researchers specializing in anaphoric reference analysis and visualization experts. The development followed, in broad terms, the design study methodology described by Sedlmair et al. (2012).

The process began with an assessment of limitations in existing tools, particularly their inability to represent anaphoric reference chains identified by multiple annotators and the resulting ambiguities. Through iterative design cycles supported by rapid prototyping, multiple designs, including matrix-based visualization approaches, were explored and subsequently discarded during both design and implementation. The final design converged on the interconnected components, visualization techniques, and visual encodings described in section 4.

Domain experts applied AmbiCoRefVis to identify ambiguous coreference cases. These usage scenarios are discussed in section 6.

4. Methods

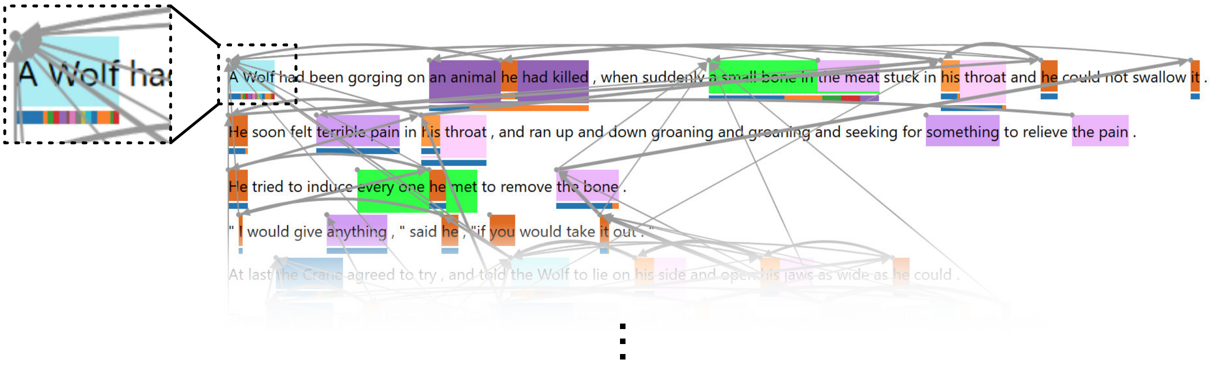
First, this section presents our tool through a concrete linguistic example. Then, we describe each visualization component in detail.

4.1. An Introductory Example

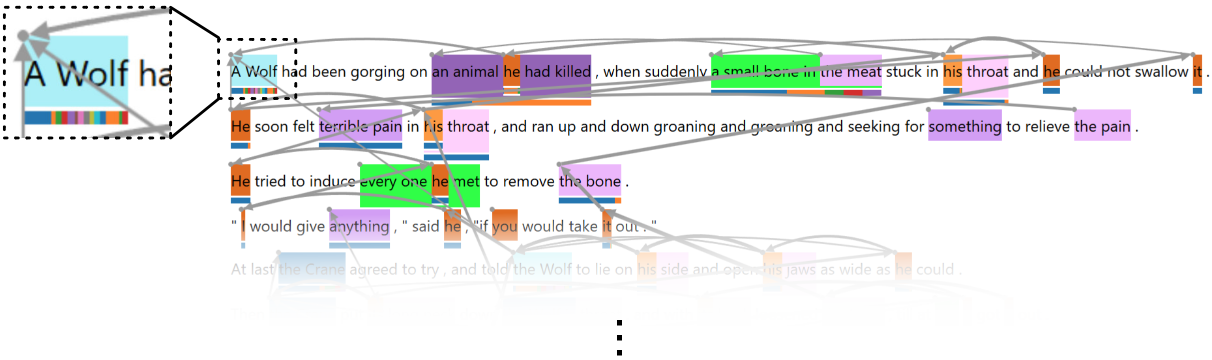
Although full coreference chains can be visualized when defined as sets as in Figure 1, MMAX2 can display alternative links for a markable only locally between individual mentions as discussed above and illustrated in Figure 3c. As a consequence, only the last mentions of the antecedents are shown and not entire competing chains. It also does not show the relationships of the antecedents or relations to subsequent mentions. Different interpretations, as well as interannotator disagreement, are not reflected either. In contrast, our tool displays competing interpretations globally through two different visualizations. In Figure 3a, we see the text component, illustrating the referring expressions within the paragraph. The graph component visualized in Figure 3b shows the tree structure of the coreference chains, highlighting nodes belonging to different chains (e.g., *his* connected with *Benjamin* and *Tommy Dreamer*). Line weights make the preferred interpretation visible immediately. Chain selection supports inspection (over the whole context).

4.2. Data Format

The *Phrase Detectives* corpus uses MAS-XML, an inline XML structure, as its data format. Each markable in the text is identified by an `<ne>` tag in the MAS-XML structure. It contains a unique ID and additional morphosyntactic features, including syntactic category, gender, number, and person. The anaphoric references identified by the players are also included as `<PDante>` tags in the MAS-XML file. For each `<ne>` tag, a `<PDante>` tag contains



(a) All references shown



(b) Only references with at least 3 annotators

Figure 2: Text component with edges filtered based on the number of players supporting a reference. A larger version of this figure is provided in the appendix.

a list of anaphoric annotations identified by the players. Each anaphoric reference identified by a player is represented by an `<anchor>` tag that specifies its antecedent and additional metadata, such as the annotation time or additional comments. For a subset of the annotations in *Phrase Detectives*, additional expert annotations using the ARRAU scheme (Uryupina et al., 2020) are available. If this information is available, we integrate it into our visualization.

4.3. Overall Design and Implementation

Our tool supports global visualization of anaphoric reference, aiming to represent an anaphoric dataset as a whole and to enable the inspection of complete coreference chains. A global perspective is essential for the analysis of variations in anaphoric chains: it allows domain experts to identify multiple competing interpretations of a given markable and discriminate whether the ambiguity is genuine or whether it is based on an annotation error (e.g., some participants overlooked the closest antecedent or a specific markable is not available for antecedent selection). To support this analysis, our visualization consists of two main components showing the text and the annotations with a varying degree of abstraction:

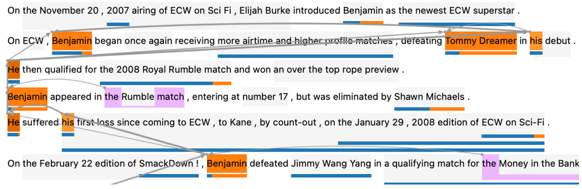
1. The text component (Figure 2) shows the underlying text with a graph overlay indicating the anaphoric references identified by the users.
2. The graph component (Figure 5, Figure 7) abstracts the underlying text and focuses on visualizing the connections between different markables.

Both our proposed main components are based on directed acyclic graphs (DAGs), where the markables are the graph's nodes and the anaphoric references are its edges. As an easy-to-use tool is one of our design goals, we designed it to be system-independent, serverless, and installation-free. We therefore developed our interactive web-based tool in JavaScript using the Svelte framework.[2] To render the DAGs, we use the graph theory and visualization library cytoscape.js (Franz et al., 2015). The code[3] and a demo[4] are available online.

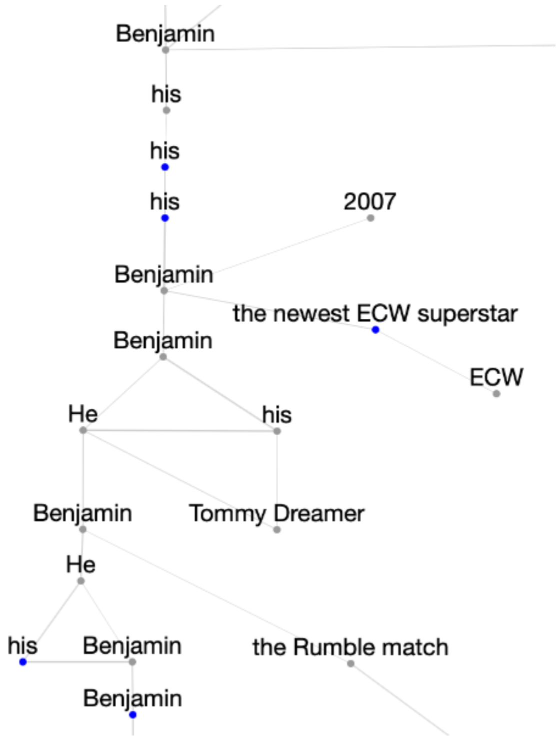
4.4. Text Component

Since the text being analyzed is the main focus of our tool, we provide a text component shown

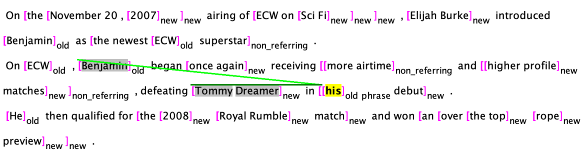
[2]https://svelte.dev
[3]https://github.com/PatrickPaetzold/AmbiCoRefVis
[4]https://graphics.uni-konstanz.de/AmbiCoRefVis/



(a) Text component with selected chain



(b) Global coreference chains visualization with graph component



(c) How the local ambiguity is visualized in MMAX

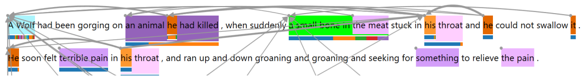
Figure 3: Visualization of ambiguity in AmbiCoRef-Vis and MMAX 2

in Figure 2. It renders the given data following the paragraphs and line breaks. Since our focus is on the text segments involved in anaphoric reference, we highlight only these markables using rectangular background shading. Each markable's background color indicates its syntactic category, as shown in the color legend on the right side of Figure 6. Users can customize these default colors as needed. When markables are nested, we visualize their hierarchy by overlaying the highlighting rectangles. For example, in Figure 6, the markable *his* is nested within the markable *his side*.

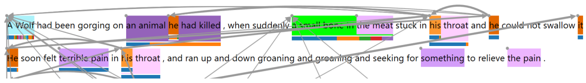
We represent anaphoric references identified by the players as a graph superimposed on the text visualization. Each node is positioned according to the location of its corresponding markable in the text, as illustrated in Figure 2. Directed edges represent the anaphoric references recognized by the players, with edge thickness indicating agreement: thinner edges correspond to references identified by only a few players, while thicker edges represent those agreed upon by many. We represent edges connecting markables on different lines as straight lines. When an edge connects markables on the same line, a horizontal edge would hinder the text readability. Thus, to reduce visual clutter, we use, in this case, rounded edges rather than straight ones.

To remove annotations that only a few players identified, and therefore are more likely annotation errors, we employ a filter to remove references supported by fewer than a specified number of players. The unfiltered visualization of all references can be cluttered, as illustrated in Figure 12a. By applying a filter to display only references supported by at least three players, the graph becomes significantly clearer and easier to interpret, as shown in Figure 12b.

Disagreement Filter The graph overlaid on the text shows ambiguity through markables referenced by multiple others, resulting in nodes with several incoming edges. This is, e.g., illustrated in the cutouts of Figure 2, where the markable *Wolf* is referenced by multiple other markables. We are primarily interested in references between markables for which the participants do not agree. Thus, we implemented a filter to hide all edges connecting markables without any player disagreement. In the graph, ambiguity is shown by markables with multiple incoming or outgoing edges.



(a) Filter to only show references to markables with at least two incoming edges



(b) Filter to only show references to markables with at least two outgoing edges

Figure 4: Filter hiding incoming and outgoing edges based on player disagreement

In Figure 4a, all edges to markables with less than two incoming edges are hidden, and in Figure 4b, all edges exiting markables with less than two outgoing edges are hidden. The markable *it* in the first line only references the markable *a small bone in the meat* and is also only referenced by another markable later on in the text. Its only in-

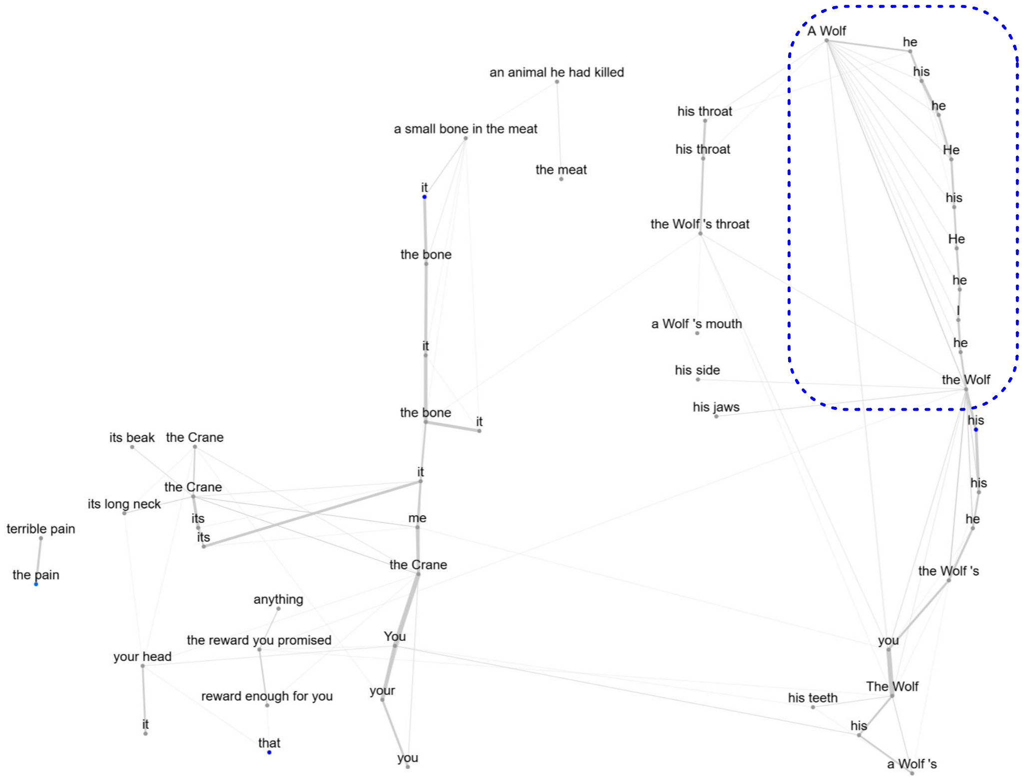


Figure 5: Graph component showing two coreference chains

coming edge is hidden by the filter in Figure 4a, while in Figure 4b, its only outgoing edge is hidden. By combining these two filters, all edges between markables with at most one incoming and outgoing edge can be hidden. Thus, only the references that the players did not agree on remain.

Double Encoding Ambiguity is expressed in the overlaid graph by multiple edges, incoming or outgoing, to a markable. In the cutouts in Figure 2, multiple edges point to the markable *Wolf*, thus it was referenced from multiple other markables in the text. In addition to the number of edges in the graph, we use a double encoding with a horizontal stacked bar chart below each markable. When applying the filters to hide edges with only little support by players, as in Figure 12b, the bars below the markable in the coutout still indicate a large number of incoming edges. Similar to the edge weight, we encode the number of players who identified a reference by the size of the bars. The bar chart can be configured to show either the distribution of edges pointing to a markable or the markables it references (i.e., the edges leaving it). Even without the edges shown in Figure 6, it is visible that the markable *the Crane* was referenced from various other markables later on in the text. We use a categorical color scheme that is unrelated to the colors used for the syntactic categories.

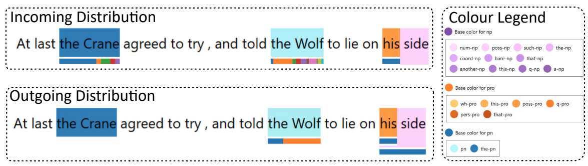
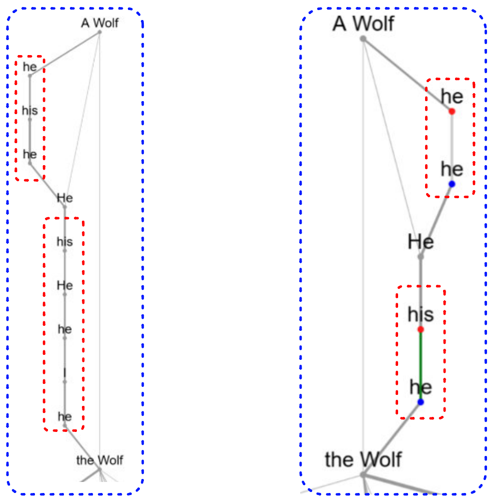


Figure 6: Encoding of syntactic category by color and double encoding of the distribution of incoming and outgoing references

Analyzing Selected Markables We implement two methods for analyzing a specific markable:
Hover interaction: When the user hovers over a markable, its incoming and outgoing references are highlighted, as seen in Figure 9a when hovering over the markable *they*. All edges are hidden, except those that enter or leave the selected markable.
Click interaction: When the user clicks on a markable, only the chains that terminate at this markable remain visible. The markables in these chains remain colored, while all others are grayed out.

4.5. Graph Component

In addition to the text component, which contains all connections above the text, we also provide a graph component. Each markable in the text is a node in the graph. All other parts of the text are not shown in this component. The anaphoric references identified by the players are shown as edges in the graph, with the edge weight encoding the number of players supporting a reference. This visualization abstracts the text, as only the relevant nodes remain. Additionally, their position is not fixed to a layout position of the markable in the text, as in the text component, but can be altered using a graph layout algorithm. We employ dagre (Franz et al., 2022) to create the force-directed graph layout shown in Figure 5. Since the node positions are defined by the graph layout, they differ significantly from those of the text component. Consequently, the graph layout is less constrained by the original text and its layout, allowing it to more effectively represent the structure formed by the references identified by the players.



(a) Filtered graph without collapsing nodes

(b) Graph simplification by collapsing nodes

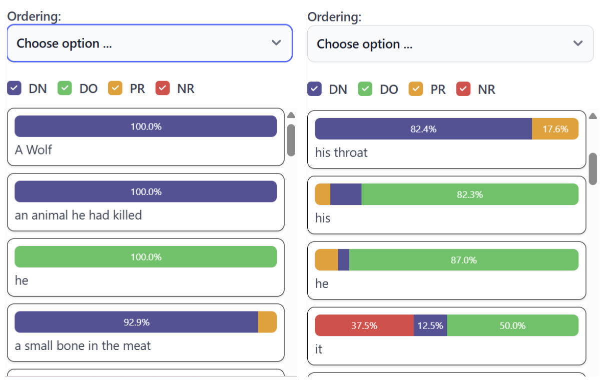
Figure 7: Collapsed edges, if they are ambiguous

In Figure 5, it is directly visible that the text–in this case, the Aesop fable *The Wolf and the Crane*–contains two main coreference chains. The edges and nodes on the left side of the figure form a chain that connects multiple mentions of the *Crane* in the fable. The nodes and edges on the right side mainly connect mentions of the *Wolf*.

As we focus on visualizing ambiguities in the dataset, linear chains as shown in Figure 7a are not particularly interesting. Therefore, we apply a node and edge collapse, removing all nodes and edges of the linear chain and connecting the beginning and end nodes with a single edge. The highlighted edges in Figure 7a are collapsed in Figure 7b. These linear chains are either directly present in the dataset when all players agree and identify the same references or arise from graph filtering, as in Figure 7. Similar to the text component, we apply a filter on the minimum number of references supporting an edge, also to the graph component. Only keeping edges supported by at least three players transforms the highlighted part of the left anaphoric reference chain shown in Figure 5 into Figure 7a, by removing slim edges.

4.6. Discourse Component

The components previously introduced only show ambiguities regarding discourse, as they indicate disagreement in the anaphoric references identified by the annotators. It is not apparent if the annotators disagree whether a markable is DN or DO, or whether it is non-referring or describes a property, i.e., predicative use. We use the stacked bar charts shown in Figure 8 to allow interacting with this information. All annotators agreed that *A Wolf* in Figure 8a is a newly appearing markable and that *he* is discourse old (referring to the wolf), while in Figure 8b the annotators were decidedly undecided about the discourse status and referentiality of *it*.

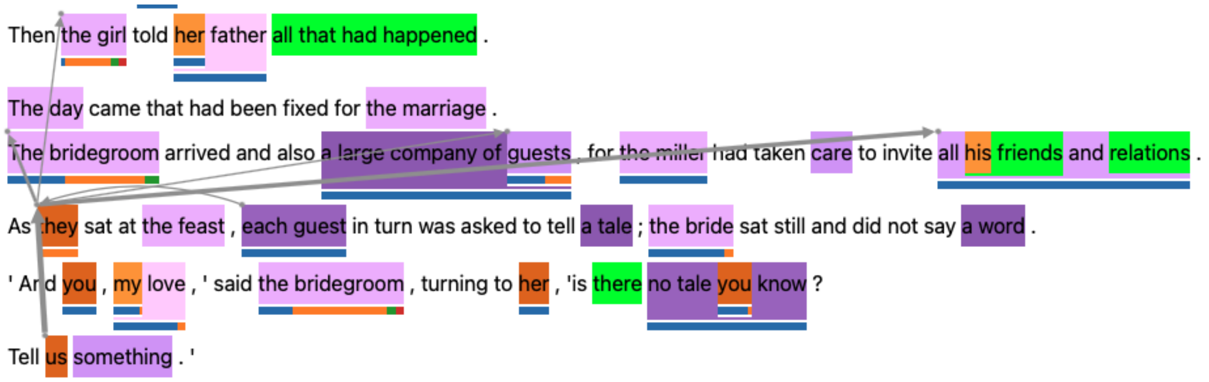


(a) (b)

Figure 8: Distribution and filter of discourse types

4.7. Linked Interactions

Each component serves a dedicated purpose. But by connecting them through interaction, new analysis possibilities emerge. To connect the individual components, we apply interaction concepts such as brushing and linking. When hovering over a markable in the text or discourse component, the corresponding node is highlighted in the graph component. To get a better overview of the possibly large graph in the graph component, it pans and zooms to the position of the markable currently hovered in the text or discourse component. Additionally, when hovering a collapsed edge in the graph component, the respective nodes and edges, which were replaced in the graph component, are highlighted in the text component. Thus, it is possible to get an overview of which parts of the graph were collapsed.



(a) Text component with selected chain

(b) Graph component

Figure 9: An example of plural anaphora

5. Conclusions and Future Work

We presented a novel interactive visualization tool to show ambiguities in anaphoric reference corpora such as *Phrase Detectives*. Its multiple components allow domain experts from linguistics to analyze the text annotations and references in a global frame, in comparison to existing annotation and analysis tools like MMAX 2. Using the provided filters, references that are only supported by a small number of annotators can be filtered. While this helps to identify ambiguities, where a large number of annotators do not agree on the references they identified, it is still up to the domain expert to decide if the ambiguity arises from an anaphoric expression that cannot be resolved in the context or just an annotation error. While this paper focused on visualizing ambiguity in anaphoric references within the *Phrase Detectives* corpus, the proposed techniques are applicable to a range of other domains. By representing NEs in the text as nodes and annotator-identified references as edges in a DAG, similar visualization approaches could be used for other sequence-oriented data, such as eye tracking studies or bioinformatics analyses.

6. Using the Visualization Tool for Ambiguity Analysis

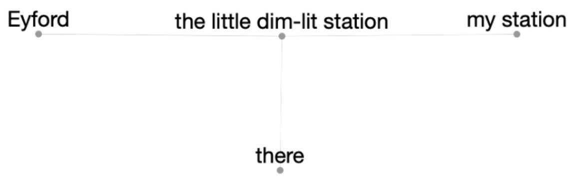
We used the tool introduced in this paper to analyze ambiguities in 65 documents in the *Phrase Detectives* corpus, release 3 (Yu et al., 2022). We briefly illustrate the type of analysis supported by the tool here.

6.1. Spurious and Incorrect Ambiguity

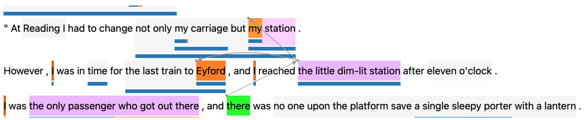
The preliminary task when analyzing anaphoric ambiguity annotation is to discriminate between genuine ambiguity, errors, and *spurious* ambiguity. Figure 3b shows how the graph component of our tool makes it easy to see and identify potential cases of ambiguity. In this example, there is a main coreference chain or references to *Benjamin*, and at least three potential cases of ambiguity: two possessive pronouns *his*, and *Tommy Dreamer*. The tool makes it easy to see that the linking of the latter to pronouns *He* and *his* is likely erroneous, as both of these markables are linked to the chain associated with *Benjamin*. The text component and graph component show that the fact that the two possessive pronouns are linked to two different antecedents is likely to be a spurious ambiguity (i.e., a case of local links to different mentions of the same entity) as all mentions the possessive pronouns are linked to are part of the same coreference chain.

6.2. A Genuine In-Context Ambiguity

After filtering spurious ambiguities, we can use the tool to investigate those cases of disagreement among subjects that reflect genuine in-context ambiguity, i.e., ambiguity that is not resolved after taking context into account (Poesio and Artstein, 2005; Versley, 2008; Recasens et al., 2011; Poesio et al., 2019). Figure 10 is an example from Arthur Conan Doyle's *The Adventure of the Engineer's Thumb* (*Phrase Detectives* 3 document `advsh12i-game`). Both the graph component and the text component show that participants chose two distinct interpretations of the referring expression *the little dim-lit station* in almost equal numbers: *Eyford* and *my station*. This ambiguity is most likely due to the fact that *Eyford* is a case of metonymy—it can be used both to refer to Eyford the village, and to Eyford station. Some participants seem to have chosen the first interpretation, while others the second one.



(a) Altered graph component

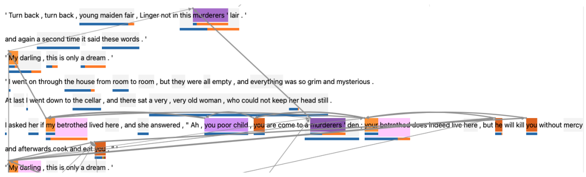


(b) Text component with selected chain

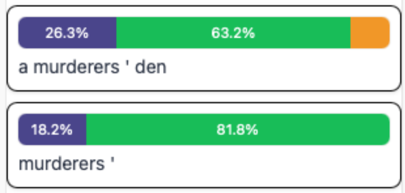
Figure 10: A genuine in-context ambiguity

6.3. DO/DN Ambiguity

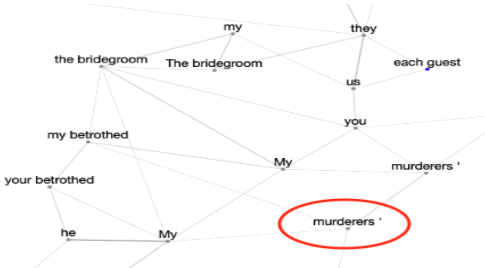
Figure 11 illustrates how the tool can be used to spot another type of ambiguity–between DO and DN interpretations. In this example (also from the *Robber Bridegroom* story), the NE *murderers'*, acting as modifier in the nominal *murderers' den*, was interpreted by the *Phrase Detectives* players in two ways: discourse-new ('a den of murderers in general') and discourse-old ('a den for the particular murderers in this story'). This is visualized in the window showing the proportion of interpretations in Figure 11b. In the second case, this mention of the murderers was also marked by 18% of the players as DO, referring to the mention of murderers in the previous NE *a murderers' lair*.



(a) Text Component grimm10aw, selected chain



(b) Distribution of interpretations for *a murderers' den* and *murderer's*: blue = DN, green = DO, orange = PR



(c) Graph component grimm10aw

Figure 11: DO vs Pred interpretation

6.4. In-Context Ambiguity with Plurals

Analyses of the *Phrase Detectives* and ARRAU corpora have confirmed the finding from Versley (2008) that many cases of in-context ambiguity originate from plural referring expressions. Our analysis of the *Phrase Detectives* corpus highlights that such cases of ambiguity can be extremely complex, as shown in Figure 9. In this example from the Grimm brothers' *The Robber Bridegroom* (*Phrase Detectives* 3 `grimm10aw`), plural *they* at the beginning of the fourth sentence was marked by the participants as having two distinct interpretations: *a large company of guests* and *all his friends and relatives*.[5] The two interpretations are reflected by two coreference chains in the graph component (Figure 9b).

[5]This is a great simplification of the number of interpretations found by the participants for this example. In *Phrase Detectives*, it is also possible to mark *split antecedent* plurals–plurals that refer to sets composed of entities introduced by multiple nominals, as in *John met Mary, then [they] went to the movies*. In total, 86 participants marked this example, finding 12 distinct interpretations, of which 9 were split-antecedent, including, e.g., the union of the two interpretations discussed in the main text, but also *the girl*, *the bridegroom*, *the miller*, *a large company of guests*, etc. We do not show all split-antecedent interpretations here, as this is another challenge in itself.

7. Acknowledgments

This work was funded in part by the German Research Foundation (DFG) Project-ID 251654672 – TRR 161 "Quantitative methods for visual computing," in part by the Dutch Research Council (NWO) through AINed Fellowship grant NGF.1607.22.002 to Massimo Poesio, "Dealing with Meaning Variation in NLP." We also wish to thank Mark-Christoph Müller for comments, suggestions, and for developing MMAX and supporting the community using it for anaphoric reference annotation for many years.

8. Bibliographical References

Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tollis. 1998. *Graph Drawing: Algorithms for the Visualization of Graphs*, 1st edition. Prentice Hall PTR.

Christin Beck, Hannah Booth, Mennatallah El-Assady, and Miriam Butt. 2020. Representation problems in linguistic annotations: Ambiguity, variation, uncertainty, error and bias. In *Proceedings of the 14th Linguistic Annotation Workshop*, pages 60–73.

Fabian Beck, Michael Burch, Stephan Diehl, and Daniel Weiskopf. 2017. A taxonomy and survey of dynamic graph visualization. *Computer Graphics Forum*, 36(1):133–159.

Georges-Pierre Bonneau, Hans-Christian Hege, Chris R. Johnson, Manuel M. Oliveira, Kristin Potter, Penny Rheingans, and Thomas Schultz. 2014. Overview and state-of-the-art of uncertainty visualization. In Charles D. Hansen, Min Chen, Christopher R. Johnson, Arie E. Kaufman, and Hans Hagen, editors, *Scientific Visualization*, pages 3–27. Springer.

Ari Bornstein, Arie Cattan, and Ido Dagan. 2020. CoRefi: A crowd sourcing suite for coreference annotation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 205–215. Association for Computational Linguistics.

David Day, Chad McHenry, Robyn Kozierok, and Laurel Riek. 2004. Callisto: A configurable annotation workbench. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC)*. European Language Resources Association.

Kees Van Deemter and Rodger Kibble. 2000. On Coreferring: Coreference in MUC and Related Annotation Schemes. *Computational Linguistics*, 26(4):629–637.

Richard Eckart de Castilho, Éva Mújdricza-Maydt, Seid Muhie Yimam, Silvana Hartmann, Iryna Gurevych, Anette Frank, and Chris Biemann. 2016. A Web-based Tool for the Integrated Annotation of Semantic and Syntactic Structures. In *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities*, pages 76–84.

Max Franz, Manfred Cheung, Harry Adel, Alexander Li, YikaJ, Ethan Keller, Sam Kortchmar, Bjørn Fjukstad, Jascha Ephraim, N8th8n8el, and Zachary Blumenfeld. 2022. cytoscape/cytoscape.js-dagre:.

Max Franz, Christian T. Lopes, Gerardo Huck, Yue Dong, Onur Sumer, and Gary D. Bader. 2015. Cytoscape.js: a graph theory library for visualisation and analysis. *Bioinformatics*, 32(2):309–311.

Pantea Haghighatkhah, Mennatallah El-Assady, Jean-Daniel Fekete, Narges Mahyar, Carita Paradis, Vasiliki Simaki, and Bettina Speckmann. 2022. Characterizing uncertainty in the visual text analysis pipeline. In *2022 IEEE 7th Workshop on Visualization for the Digital Humanities*, pages 25–30.

Chris R. Johnson. 2004. Top scientific visualization research problems. *IEEE Computer Graphics and Applications*, 24(4):13–17.

Maurice Koch, Nan Cao, Daniel Weiskopf, and Kuno Kurzhals. 2025. Active gaze labeling: Visualization for trust building. *IEEE Transactions on Visualization and Computer Graphics*, 31(7):3942–3956.

Olga Krasavina and Christian Chiarcos. 2007. PoCoS - Potsdam Coreference Scheme. In *Proceedings of the Linguistic Annotation Workshop*, pages 156–163. Association for Computational Linguistics.

Alan M. MacEachren, Robert E. Roth, James O'Brien, Bonan Li, Derek Swingley, and Mark Gahegan. 2012. Visual semiotics & uncertainty visualization: An empirical study. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2496–2505.

Mark-Christoph Müller and Michael Strube. 2006. Multi-level annotation of linguistic data with MMAX2. In Sabine Braun, Kurt Kohn, and Joybrato Mukherjee, editors, *Corpus Technology and Language Pedagogy. New Resources, New Tools, New Methods*, volume 3 of *English Corpus Linguistics*, pages 197–214. Peter Lang.

Christoph Müller and Michael Strube. 2003. Multi-Level Annotation in MMAX. In *Proceedings of the Fourth SIGdial Workshop of Discourse and Dialogue*, pages 198–207.

Bruno Oberle. 2018. SACR: A Drag-and-Drop Based Tool for Coreference Annotation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*. European Language Resources Association.

Maciej Ogrodniczuk, Katarzyna Głowińska, Mateusz Kopeć, Agata Savary, and Magdalena Zawisławska. 2013. The Polish Coreference Corpus. In *Proceedings of Language and Technology Conference*, pages 215–226. Springer International Publishing.

Constantin Orăsan. 2003. PALinkA: A highly customisable tool for discourse annotation. In *Proceedings of the Fourth SIGdial Workshop of Discourse and Dialogue*, pages 39–43.

Lace Padilla, Matthew Kay, and Jessica Hullman. 2021. *Uncertainty Visualization*, pages 1–18. John Wiley & Sons, Ltd.

Alex Pang, Craig M. Wittenbrink, and Suresh K. Lodha. 1997. Approaches to uncertainty visualization. *The Visual Computer*, 13(8):370–390.

Massimo Poesio and Ron Artstein. 2005. The Reliability of Anaphoric Annotation, Reconsidered: Taking Ambiguity into Account. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pages 76–83. Association for Computational Linguistics.

Massimo Poesio, Maris Camilleri, Paloma Carretero Garcia, Juntao Yu, and Mark-Christoph Müller. 2024. The ARRAU 3.0 Corpus. In *Proceedings of the 5th Workshop on Computational Approaches to Discourse*, pages 127–138.

Massimo Poesio, Jon Chamberlain, Silviu Paun, Juntao Yu, Alexandra Uma, and Udo Kruschwitz. 2019. A crowdsourced corpus of multiple judgments and disagreement on anaphoric interpretation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 1778–1789. Association for Computational Linguistics.

Massimo Poesio, Nils Diewald, Maik Stührenberg, Jon Chamberlain, Daniel Jettka, Daniela Goecke, and Udo Kruschwitz. 2012. Markup infrastructure for the anaphoric bank: Supporting web collaboration. In Alexander Mehler, Kai-Uwe Kühnberger, Henning Lobin, Harald Lüngen, Angelika Storrer, and Andreas Witt, editors, *Modeling, Learning, and Processing of Text Technological Data Structures*, pages 175–195. Springer Berlin Heidelberg.

Massimo Poesio and Uwe Reyle. 2001. Underspecification in anaphoric reference. In *Proceedings of the Fourth International Workshop on Computational Semantics*, pages 286–300.

Massimo Poesio, Patrick Sturt, Ron Arstein, and Ruth Filik. 2006. Underspecification and anaphora: Theoretical issues and preliminary evidence. *Discourse Processes*, 42(2):157–175.

Ellen F. Prince. 1992. The ZPG letter: subjects, definiteness, and information status. In S. Thompson and W. Mann, editors, *Discourse description: diverse analyses of a fund-raising text*, pages 295–325. John Benjamins.

Marta Recasens, Eduard Hovy, and M. Antònia Martí. 2011. Identity, non-identity, and near-identity: Addressing the complexity of coreference. *Lingua*, 121(6):1138–1152.

Marta Recasens, M. Antonia Martí, and Constantin Orasan. 2012. Annotating near-identity from coreference disagreements. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation*, pages 165–172.

Nils Reiter. 2018. CorefAnnotator - A New Annotation Tool for Entity References. In *Abstracts of EADH: Data in the Digital Humanities*.

Nils Reiter. 2025. nilsreiter/CorefAnnotator. `https://github.com/nilsreiter/CorefAnnotator`.

Dominik Sacha, Hansi Senaratne, Bum Chul Kwon, Geoffrey P. Ellis, and Daniel A. Keim. 2016. The role of uncertainty, awareness, and trust in visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):240–249.

Christoph Schulz, Arlind Nocaj, Jochen Görtler, Oliver Deussen, Ulrik Brandes, and Daniel Weiskopf. 2017. Probabilistic graph layout for uncertain network visualization. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):531–540.

Michael Sedlmair, Miriah Meyer, and Tamara Munzner. 2012. Design Study Methodology: Reflections from the Trenches and the Stacks. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2431–2440.

Maik Stührenberg, Daniela Goecke, Nils Diewald, Alexander Mehler, and Irene Cramer. 2007. Web-based annotation of anaphoric relations and lexical chains. In *Proceedings of the Linguistic Annotation Workshop*, pages 140–147. Association for Computational Linguistics.

Olga Uryupina, Ron Artstein, Antonella Bristot, Federica Cavicchio, Francesca Delogu, Kepa Rodriguez, and Massimo Poesio. 2020. Annotating a broad range of anaphoric phenomena, in a variety of genres: the ARRAU corpus. *Journal of Natural Language Engineering*, 26(1).

Yannick Versley. 2008. Vagueness and Referential Ambiguity in a Large-Scale Annotated Corpus. *Research on Language and Computation*, 6(3-4):333–353.

Luis von Ahn. 2006. Games with a purpose. *Computer*, 39(6):92–94.

Daniel Weiskopf. 2022. Uncertainty visualization: Concepts, methods, and applications in biological data visualization. *Frontiers in Bioinformatics*, Volume 2 - 2022.

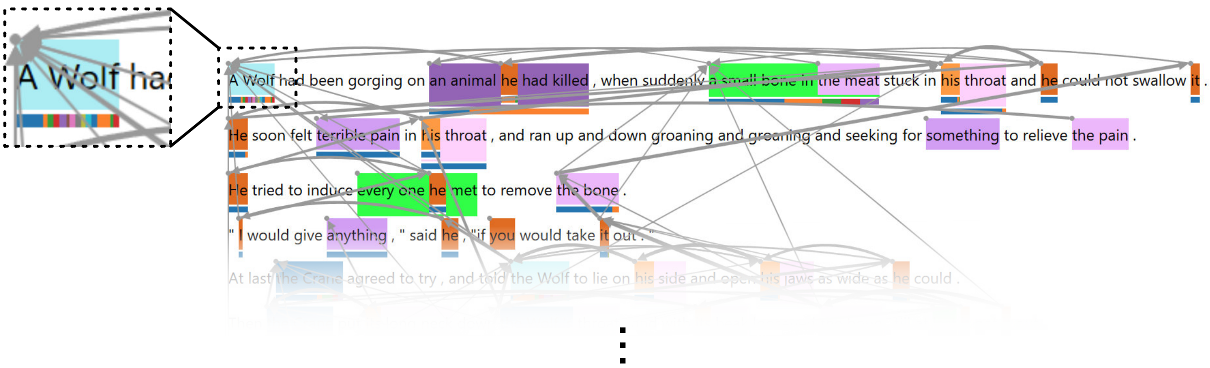
Antoine Widlöcher and Yann Mathet. 2012. The Glozz platform: a corpus annotation and mining tool. In *Proceedings of the 2012 ACM symposium on Document engineering*, pages 171–180. ACM.

Bingyang Ye, Jingxuan Tu, and James Pustejovsky. 2023. Scalar anaphora: Annotating degrees of coreference in text. In *Proceedings of the Sixth Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 28–38. Association for Computational Linguistics.

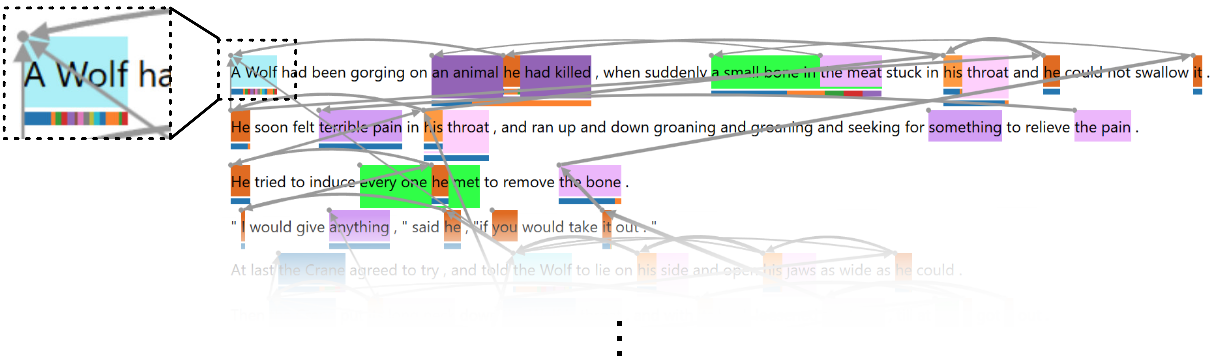
Juntao Yu, Silviu Paun, Maris Camilleri, Paloma Carretero Garcia, Jon Chamberlain, Udo Kruschwitz, and Massimo Poesio. 2022. Aggregating Crowdsourced and Automatic Judgments to Scale Up a Corpus of Anaphoric Reference for Fiction and Wikipedia Texts.

Amir Zeldes. 2022. Opinion Piece: Can we Fix the Scope for Coreference?: Problems and Solutions for Benchmarks beyond OntoNotes. *Dialogue & Discourse*, 13(1):41–62.

A. Appendix



(a) All references shown



(b) Only references with at least 3 annotators

Figure 12: Text component with edges filtered based on the number of players supporting a reference

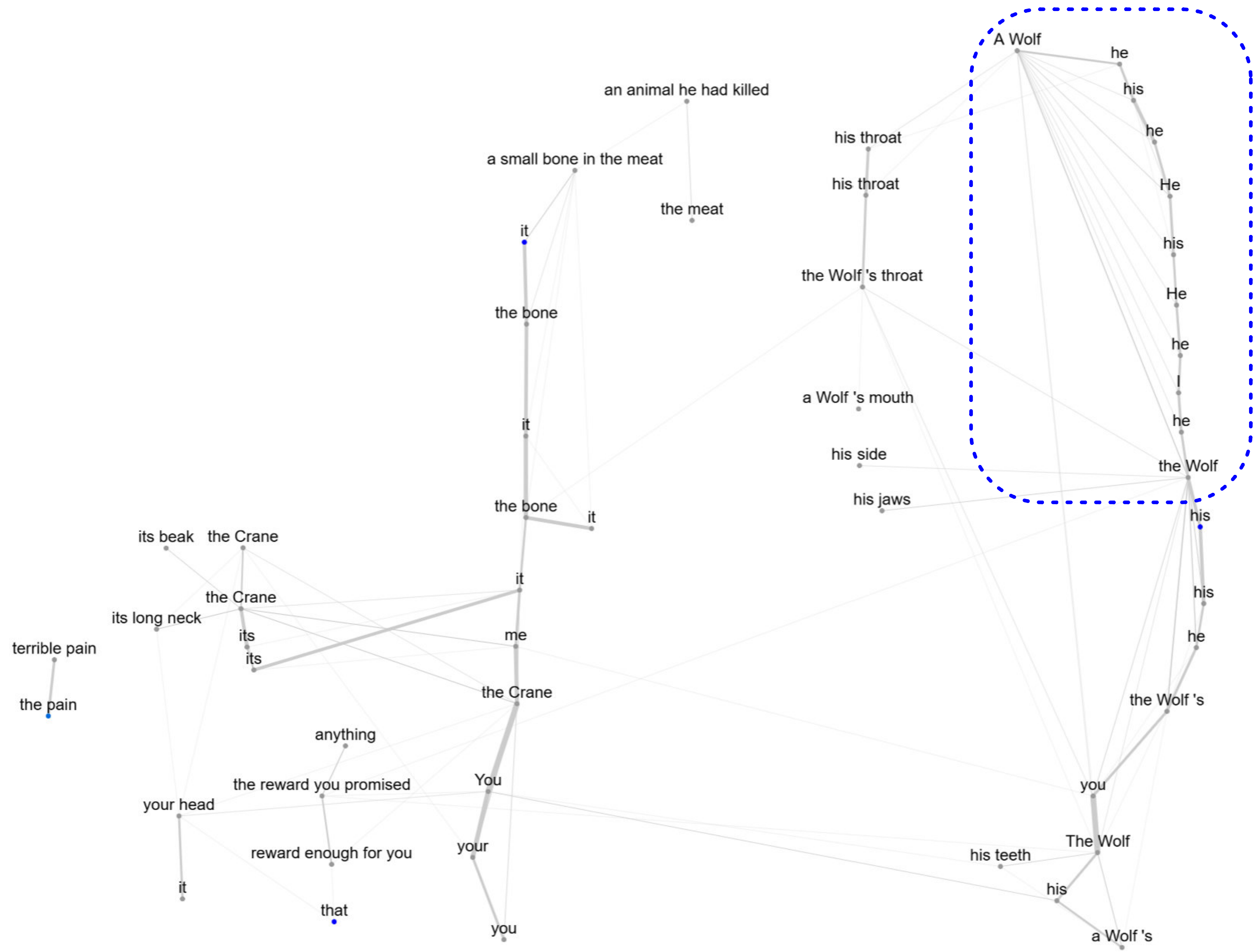


Figure 13: Graph component showing two coreference chains