CORVIDAE: Coreference Resolution Visual Development & Analysis Environment

Nico Möller
Institute of Cognitive Science
University of Osnabrück
49069 Osnabrück
Germany
nico.moeller@uni-osnabrueck.de

Gunther Heidemann
Institute of Cognitive Science
University of Osnabrück
49069 Osnabrück
Germany
gunther.heidemann@uni-osnabrueck.de

ABSTRACT

Communication whether in verbal or written form is part of our daily life. Hence, we as humans have developed a set of skills that enable us to follow a discourse and extract important information from a text quite easily. For a machine, however language understanding is a quite challenging problem and considered to be AI-complete, i.e. a machine must reach human level intelligence in order to solve this task. Recent developments, especially those forming the semantic web, offer new ways of incorporating world knowledge into natural language processing methods, while at the same time progress made in the latter will help pushing the dream of the global knowledge graph closer to reality. In this paper we present CORVIDAE (Coreference Resolution Visual Development & Analysis Environment) a tool for NLP developers to analyse and eventually improve coreference resolution algorithms specially designed for those that interact with world knowledge.

1. INTRODUCTION

Coreference resolution (CR) is a subtask of information extraction and describes the task of identifying all mentions in a given document and group those together that refer to the same entity [18]. CR is one of the core tasks in information extraction, making it a necessary preprocessing step before other algorithms can be applied. It has been an active field of research since the 1960s. Whereas research in the early years of CR was dominated by heuristic approaches built on computational theories of discourse [5, 6, 24], methods based on machine learning became more and more popular due to the broader availability and increased computational power in the 1990s. Most common methods are based on supervised learning, using string-matching, syntactic, grammatical or semantic features on those mentions. Observing the course of development in this field, a trend becomes visible that starts with local features [1, 16] and goes on towards more global models [14, 22]. The next logical step would be to go beyond global features, i.e. incorporating pieces of information that are not in the document, but can help to solve this task. This includes semantic relatedness features extracted from knowledge bases like *WordNet*, *Wikipedia* or *YAGO* that already have proven to be valuable additions [20, 23]. Additionally, there have been approaches to solve subtasks in information extraction like coreference resolution and named entity linking in a joint fashion rather than separately [13, 9]. An elaborated error analysis of the state-of-the-art Stanford *CoreNLP* system has shown that 41.7% of errors can be attributed to the lack of background knowledge of the system [12]. Another motivation behind this shift can be found in the increased interest in information extraction and analysis in the recent years. Besides *Big Data*, another keyword that kept appearing in the recent years is the one of the *semantic web* [3]. The goal of the semantic web is to increase the exchangeability of data as well as its usability. Web documents should be tagged with additional information that set a context for this document creating a machine-readable knowledge-graph that contains information about persons, organisations, places or events mentioned in the text as well as their connections to other entities. Without proper background knowledge, it is impossible to integrate extracted information correctly into an existing knowledge base. Taking the outlook on data production[1] into consideration as well as the fact that only 4 million out of 175 million active domains [7] use the semantic markup on their websites[2], it seems a good idea to work on increasing the quality of coreference resolution systems as these play a crucial role in solving problems currently encountered in *Big Data* analysis and fulfilling the dream of the *semantic web*.

2. RELATED WORK

Tools for visualising coreference annotation data can roughly be divided into two groups. The first group of tools focuses on the annotation itself with the aim of creating data that can be used as training input for NLP algorithms like coreference resolution. Most popular along these are *MMAX2* [17], *PAlinkA* [19] and *BRAT* [25]. However, those are mainly text-based with only a very limited capacity of visualising data besides a few visual cues like highlighting mention groups or showing links between mentions. Another way of visualising coreference data was introduced by the *TrEd* annotator using trees to visualise coreference as well as other tree based annotations [8]. The *SUCRE* project in contrast utilised self organising maps to visualise coref-

[1] According to a recent study conducted by the EMC as part of their *Digital Universe* Series, humanity is currently producing about 4.4 Zettabytes of data, which will tenfold by 2020

[2] `http://news.netcraft.com/archives/2015/10/16/october-2015-web-server-survey.html`

SEMANTICS 2016: Posters and Demos Track
September 13-14, 2016, Leipzig, Germany

erence features [4]. Additionally, human annotators should be provided with suggestions for possible coreferences in a semi-supervised fashion to speed up the annotation process.

Exploring already annotated data can be done with those tools, but due to their intended purpose, they lack important features that are needed for error analysis. Crucial would be the capability of comparing a data set against a gold standard annotation.

Tools that focus on the NLP developer, on the other hand, are quite rare. A widely used tool kit for error analysis in coreference systems is that of Kummerfeld & Klein [11]. Their approach utilised transformation operations to automatically categorise errors in the output of coreference systems, but also lacks any functionality to visualise their results. Kuhn et al.[10] presented the *ICARUS Coreference Explorer (ICE)*, a framework specially designed to provide tools for visualisation, search and error analysis for coreference annotations. Besides a tree view similar to *TrEd*, it utilised the entity grid [2], a tabular view of entities in a document to give both a summed up view of mentioned entities as well as show changes of entity descriptions throughout the document. *ICE* however, is focused on the links between pairs of links, neglecting global features on groups of mentions and features beyond that. Complementing those is the tool from Martschat et al. [15], which provides a text-based visualisation similar to *BRAT*. Although the functionality to add world knowledge is mentioned, the system is not yet suitable to handle analysis on the output of cross-document coreference resolution or entity linking systems. To solve those problems we present *CORVIDAE* a tool for the analysis and development of coreference resolution systems that incorporate world knowledge.

3. CORVIDAE

CORVIDAE is a web-based application. The backend is written in *Scala*[3], built upon the the *Play* web application framework[4]. *HTML5* and *JavaScript* built the basis for the frontend, which uses the *BRAT* library [5] as well as the *D3.js JavaScript* library[6] for interactive visualisations.

As it is a tool intended for the error analysis in coreference resolution systems, we had to choose one such a system as a starting point. Our choice fell on the the Stanford *CoreNLP* system respectively its extensions that incorporate named entity linking [21, 13, 9]. The coreference resolution system uses a multi-pass sieve that extracts features tier-wise with decreasing level of precision in a hierarchical fashion. Stronger features are applied with a higher precedence than weaker features, each tier building upon results from previous ones. Due to the systems architecture, it is easy to grab intermediate results for a detailed analysis, which makes it an ideal starting point for our analysis tool. *CORVIDAE* however, can also be extended to interact with other coreference systems.

3.1 Workflow

CORVIDAE takes preprocessed text documents as an input. On the one hand, those get annotated by humans to produce gold standard results. The same text documents

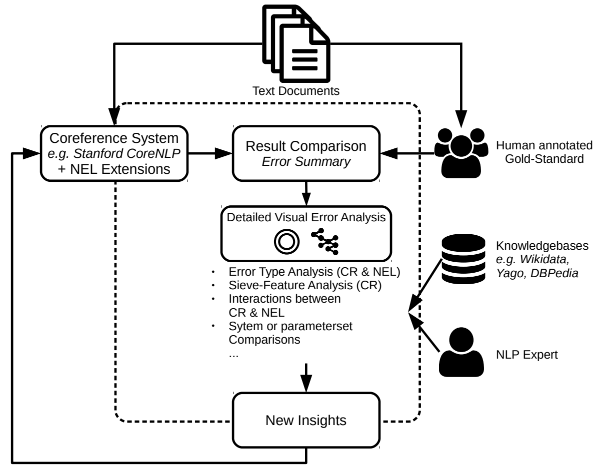

Figure 1: The workflow of the system.

are also automatically processed by a CR system, e.g. the aforementioned Stanford CoreNLP system with NEL extensions [21, 13, 9]. In an initial step, the user is presented with an overview of the CR systems performance, while additionally pointing out different types of errors the system has produced[7]. Following this step the user can then use *CORVIDAE*s different visualisation modes for a variety of tasks. The modes can be used in both a consecutive as well as a parallel manner to provide further insights and eventually lead to the elimination of the sources of errors.

A typical use case would be to focus on a certain type of error produced by the system, which works for both the CR as well as NEL part. Depending on which part has been chosen by the user we can for example use either a mention centred or entity centred visualisation to analyse one (or several) documents thoroughly in order to gain insights on what caused the error. Those insights will help to modify existing parameters of our initial system or develop completely new ones. Afterwards, the system has to be run again to compare the new performance against the old one. This step can be repeated as often as necessary.

In the following subsection we will have a closer look at the different visualisation modes the *CORVIDAE* framework has to offer.

3.2 Visualisations

Besides the text-based visualisation provided by the *BRAT* library, *CORVIDAE* offers a few other view modes on coreference annotation data. In the following subsections we present a few of *CORVIDAE*s visualisation modes that focus on different parts of the error analysis. All of these visualisation modes are types of circular layouts, a compact drawing style for information visualisation that is especially popular in the area of bioinformatics.

3.2.1 Radial Network Diagram

Figure 2 depicts an example of a radial network diagram. Shown on the outer rim are the found mentions within a

[3] http://www.scala-lang.org/

[4] https://www.playframework.com/

[5] http://brat.nlplab.org/embed.html

[6] https://d3js.org/

[7] For more details on the different error types produced, have a look at the analysis section of [21, 13, 9]

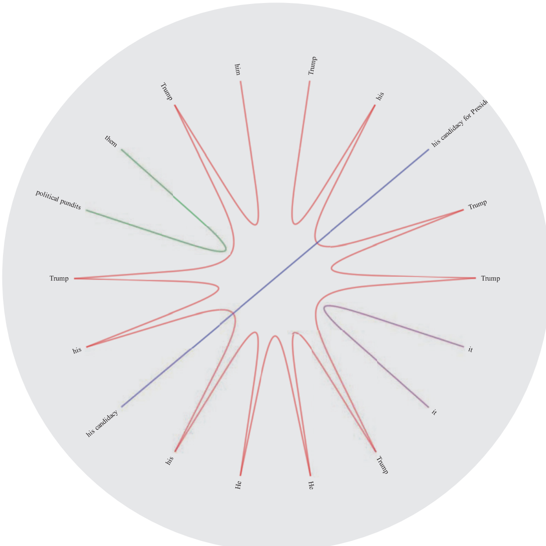

Figure 2: A radial network diagram, showing links between different mentions within a text. Links are colour-coded according to cluster membership.

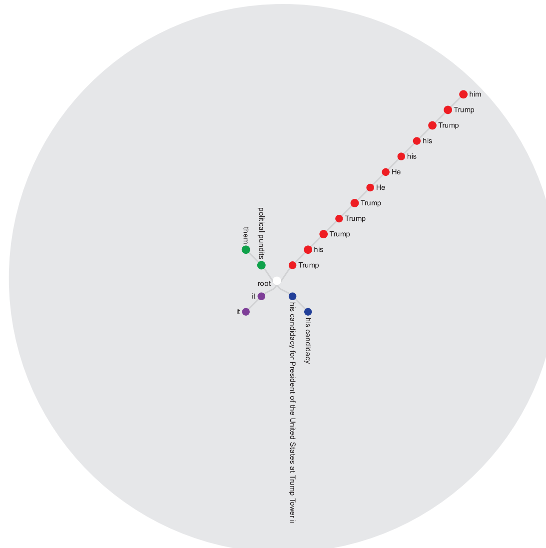

Figure 3: A radial directed graph diagram, showing a tree based coreference annotation. Nodes are colour-coded according to cluster membership.

document, currently in the order of appearance within the document. Entity clusters are depicted by colour coding. Arcs connecting two mentions indicate a coreference between them. The mentions can also be sorted and split into their corresponding entity clusters for further inspection of the individual mentions. As mentioned before the *D3.js* library allows for interactivity, henceforth the visualisations allow for highlighting via hovering or filtering via queries, as well as displaying additional information like the linked real world entity when selecting an entity cluster.

In order to compare two annotation results, the computed diagrams can be overlayed for an easy spotting of differences and therefore potential errors.

3.2.2 Radial Directed Graph Diagram

The radial directed graph diagram has been incorporated in two different modes.

Mention centred: This mode allows for the visualisation of tree based coreference annotations similar to the ones found in *TrEd* or *ICE*, but instead of a triangular layout we are using a radial one, which allows for a much more compact and cleaner representation. Originating from the inner document root, nodes in the tree correspond to mentions in the text, whereas links indicate coreference between those. Each branch from the root node corresponds to a cluster representing an entity. Figure 3 shows the same annotation result as figure 2, this time using a radial tree layout.

Entity centred: The second mode is concerned with the visualisation of semantic background knowledge. It can visualise information extracted from the documents itself, but is not solely restricted to it. Named entity linking usually uses a knowledge base that serves as an anchor. These can be exploited to provide additional context for the NLP developer, as well as to evaluate and compare extracted information against the knowledge base. The colour of the links indicates that no (orange), supporting (green) or contradict-

Figure 4: A radial directed graph diagram, showing entities found within a document. Additional links and entities have been provided by querying a knowledge base.

ing (red) information has been found in the knowledge base. For example, if a was-born-in relation between entity a and entity b is mentioned within a sentence and this fact can be found in our database the link would be green, if no relation can be found the link would be orange and red in case contradicting information has been found. The size of the dots corresponds to the number of in and outgoing edges. An example showing this can be found in figure 4.

The same overlay technique as mentioned in section 3.2.1 can also be used on this type of visualisation. Mention centred this view allows to compare different sets of annotations for one document, whereas the entity centred view can be used to check results of one cross-document coreference resolution system over two documents.

3.2.3 Radial Sequence Diagram

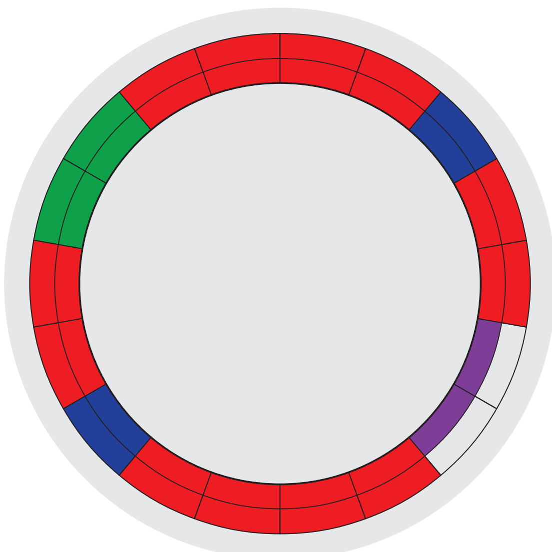

Figure 5: A radial sequence diagram, comparing the annotation (found mentions, colour-coded according to cluster membership) from a CR system (inner sequence) against a gold standard annotation (outer sequence).

Originally used to compare genome sequences, we utilise this technique to quickly compare different annotation results. In a first step, the user is prompted to specify a feature he wants to analyse. A simple example would be cluster membership, but ultimately any part of the annotation data we have access to can be used. Afterwards, this feature is encoded into a colour sequence. Figure 5 shows a simple comparison between the result of a CR system and a gold standard annotation. Due to its compact design, this view mode allows not only to compare two annotation results to one another but even multiple ones.

This view mode, besides being the most versatile one, is the key feature of *CORVIDAE*. We can use it to:

- compare different configurations for a single coreference system or results from different systems on a single document,
- analyse the propagation of errors in the multi-sieve model level wise,
- compare different documents to find out if they get linked to the same entities.

These are just a few examples how *CORVIDAE* can support the NLP developer to debug their system.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we presented *CORVIDAE* a tool designed for NLP developers for the visual error analysis of cross-document coreference systems. Besides a text-based display mode, this tool offers a variety of circular visualisations to display coreference annotation data, which will help to analyse and debug cross-document coreference resolution algorithms. In its current state *CORVIDAE* supports three different circular visualisations, namely:

- radial network diagrams,
- radial directed graph diagrams,
- radial sequence diagrams.

Each is intended to support the NLP developer in tracking down, isolating and locating errors caused by the CR system. All of these visualisations are interactive and highly customizable, making it easy for the user to adapt the system to his needs. As a starting point for our analysis, we choose the state-of-the-art *CoreNLP* CR system, but *CORVIDAE* can easily be extended to support other systems as well.

CORVIDAE is a work in progress and still under development. For more information you can visit the projects website[8], where you can find more details on the systems architecture and intended workflow, as well as information concerning the different visualisation modes and usage examples. An online demonstration of the system is expected to be added soon.

Future works include a thorough evaluation of *CORVIDAE* in cooperation with NLP Experts to validate the systems benefits. Eventually, those results will flow back into the development of analysis routines, that can be used as a template for certain types of errors as well as additional modes of visualisation, in the case of lacking functionality or missing practicality of existing ones.

5. REFERENCES

[1] C. Aone and S. W. Bennett. Evaluating automated and manual acquisition of anaphora resolution strategies. In *Proceedings of ACL 1995*, pages 122–129. Association for Computational Linguistics, 1995.

[2] R. Barzilay and M. Lapata. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34, 2008.

[3] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5):34–43, 5 2001.

[4] A. Burkovski, W. Kessler, G. Heidemann, H. Kobdani, and H. Schütze. Self organizing maps in nlp: Exploration of coreference feature space. In *Proceedings of the 8th International Conference on Advances in Self-organizing Maps*, WSOM'11, pages 228–237, Berlin, Heidelberg, 2011. Springer-Verlag.

[8] https://ikw.uos.de/%7Ecv/publications/SEMANTICS16

[5] B. J. Grosz. The representation and use of focus in a system for understanding dialogs. In *Proceedings of IJCAI 1977 - Volume 1*, IJCAI'77, pages 67–76, San Francisco, CA, USA, 1977. Morgan Kaufmann Publishers Inc.

[6] B. J. Grosz, A. K. Joshi, and S. Weinstein. Providing a unified account of definite noun phrases in discourse. In *Proceedings of ACL 1983*, pages 44–50. Association for Computational Linguistics, 1983.

[7] R. V. Guha. Light at the end of the tunnel. Talk at the 12th International Semantic Web Conference (ISWC), Sydney, 10 2013.

[8] J. Hajic, B. Vidová-Hladká, and P. Pajas. The prague dependency treebank: Annotation structure and support. In *Proceedings of the IRCS Workshop on Linguistic Databases*, pages 105–114, 2001.

[9] H. Hajishirzi, L. Zilles, D. S. Weld, and L. S. Zettlemoyer. Joint coreference resolution and named-entity linking with multi-pass sieves. In *EMNLP*, pages 289–299. ACL, 2013.

[10] J. Kuhn, M. Gärtner, A. Björkelund, G. Thiele, and W. Seeker. Visualization, search, and error analysis for coreference annotations. *ACL 2014*, page 7, 2014.

[11] J. K. Kummerfeld and D. Klein. Error-driven analysis of challenges in coreference resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 265–277, Seattle, Washington, USA, October 2013.

[12] H. Lee, A. Chang, Y. Peirsman, N. Chambers, M. Surdeanu, and D. Jurafsky. Deterministic Coreference Resolution Based on Entity-Centric, Precision-Ranked Rules. *Computational Linguistics*, 39(4):885–916, dec 2013.

[13] H. Lee, M. Recasens, A. Chang, M. Surdeanu, and D. Jurafsky. Joint entity and event coreference resolution across documents. In *Proceedings of EMNLP-CoNLL 2012*, EMNLP-CoNLL '12, pages 489–500, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[14] X. Luo, A. Ittycheriah, H. Jing, N. Kambhatla, and S. Roukos. A mention-synchronous coreference resolution algorithm based on the bell tree. In *Proceedings of ACL 2014*, page 135. Association for Computational Linguistics, 2004.

[15] S. Martschat, T. Göckel, and M. Strube. Analyzing and visualizing coreference resolution errors. In *Proceedings of the 2015 Conference of the North Americal Chapter of the Association for Computational Linguistics: Demonstration Session,* Denver, Col., 31 May – 5 June 2015, pages 6–10, 2015.

[16] J. F. McCarthy and W. G. Lehnert. Using decision trees for coreference resolution. In *Proceedings of IJCAI 1995*, pages 1050–1055, 1995.

[17] C. Müller and M. Strube. Multi-level annotation of linguistic data with MMAX2. In S. Braun, K. Kohn, and J. Mukherjee, editors, *Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods*, pages 197–214. Peter Lang, Frankfurt a.M., Germany, 2006.

[18] V. Ng. Supervised Noun Phrase Coreference Research: The First Fifteen Years. *Acl*, (July):1396–1411, 2010.

[19] C. Orăsan. PALinkA: a highly customizable tool for discourse annotation. In *Proceedings of the 4th SIGdial Workshop on Discourse and Dialog*, pages 39 – 43, Sapporo, Japan, July, 5 -6 2003.

[20] S. P. Ponzetto and M. Strube. Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 192–199. Association for Computational Linguistics, 2006.

[21] 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 EMNLP 2010*, EMNLP '10, pages 492–501, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[22] A. Rahman and V. Ng. Supervised models for coreference resolution. In *Proceedings of EMNLP 2009: Volume 2-Volume 2*, pages 968–977. Association for Computational Linguistics, 2009.

[23] A. Rahman and V. Ng. Coreference resolution with world knowledge. In *Proceedings of ACL 2011: Human Language Technologies - Volume 1*, HLT '11, pages 814–824, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[24] C. L. Sidner. Towards a computational theory of definite anaphora comprehension in english discourse. Technical report, Cambridge, MA, USA, 1979.

[25] P. Stenetorp, S. Pyysalo, G. Topić, T. Ohta, S. Ananiadou, and J. Tsujii. brat: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations Session at EACL 2012*, Avignon, France, April 2012. Association for Computational Linguistics.