# Integrating user-specified Knowledge for semi-automatic Coreference Resolution

**Schmidt, David**
david.b.schmidt@uni-wuerzburg.de
Universität Würzburg, Deutschland

**Krug, Markus**
markus.krug@uni-wuerzburg.de
Universität Würzburg, Deutschland

**Puppe, Frank**
frank.puppe@uni-wuerzburg.de
Universität Würzburg, Deutschland

## Introduction

Coreference Resolution is a challenge which has seen the interest of researchers for half a century, but at the current point in time, even state-of-the-art algorithms (Joshi et al., 2019)cannot produce reliable results when applied in a fully automatic manner. The main problem of an automatic coreference resolution system is that decisions that occur late in the text can heavily influence prior decisions and so the resulting clustering and its composition is hard to understand. We found that, when applying the end-to-end coreference resolution algorithm (Lee et al., 2017), the algorithm tends to reset its understanding of the text every couple of paragraphs, which results in a mixture of grave errors when aggregated over the document. Coreference resolution is a necessary and very important step however, when analysing the content of a literary text. Different engines detect knowledge on a local level and coreference resolution aggregates this knowledge to the document scope of the text. This means that, without a reliable coreference resolution module, most applications that require an aggregated view of the texts (which is very important for most distant reading experiments) cannot be researched efficiently. In this work, we try to overcome the challenge of coreference resolution by presenting a semi-automatic mechanism instead of a fully automatic system. This mechanism allows the users to integrate their prior knowledge about characters of a literary text and their relations. We do this by parsing this knowledge into a machine-readable data structure and integrate this knowledge into our rule-based coreference resolution system, which was extended from (Krug et al., 2015).

## Related Work

There have been various works trying to integrate world knowledge into coreference resolution. Ng (2007) added several new features to a coreference resolution system based on machine-learning, e.g. a feature for the semantic similarity of mentions and a feature based on patterns extracted from the training data. Others used knowledge that was extracted from knowledge bases like YAGO (Suchanek et al., 2007), among them Bryl et al. (2010), who model synonyms and mention types using logical constraints in a Markov Logic Network (Richardson and Domingos, 2006), and Rahman and Ng (2011), who extract relation triples between two mentions and convert them into features for their algorithm.

Aralikatte et al. (2019) use knowledge bases in a slightly different way. They apply the neural coreference resolution system of Lee et al. (2018) in a reinforcement learning setting and reward it, the more valid relations can be extracted from it. A valid relation is evaluated against knowledge bases like Wikidata and Wikipedia.

Unfortunately, these approaches are impractical for historic novels since the knowledge bases used there mostly contain knowledge about real-world entities while the novels deal with fictional characters.

## Method

Our approach basically allows the user to model the knowledge which Wikipedia or Wikidata contain about real-world entities for fictional characters. The knowledge is represented as character sheets. For each character, the user needs to determine a unique name (e.g. Richard Landsfeld) and can optionally provide a first name (Richard), last name (Landsfeld), the character's gender (male), a list of strings which are used as synonyms for the character (e.g. Baron von Landsfeld) and a list of strings which do not refer to this character. In addition to that, the user can specify the character's relations to other characters by providing the name of the other character and a list of possible labels for this character (e.g. Lydia - Ehefrau/ Gattin).

The knowledge provided by the user is used in several different sieves of the algorithm. First name, last name and gender are used in a sieve which already existed prior to this work. It uses first and last names to merge clusters if they are compatible with respect to several other meta data fields (like gender). For this work, it was expanded to also merge clusters which contain mentions that have been identified to belong to the same character from user-specified knowledge. This identification is done in one of the first sieves by comparing the mentions' texts to a character's unique name, first name, synonyms and, if no other character has the same last name, last name. Mentions that have been identified to belong to a character this way are from then on prevented from being merged into a cluster

together with mentions belonging to other characters as well as mentions which have one of the character's non-coreferent strings as their text.

The relations contained in the user-specified knowledge are used in one of the last sieves of the algorithm. It looks for constructions where one mention is (a) a possessive, demonstrative or relative pronoun used as an attribute, (b) a genitive or (c) preceded by a token 'von' which means that one mention is a prepositional modifier of another, like 'seine Ehefrau', 'Richards Ehefrau' or 'Ehefrau von Richard'. These constructions can be used if the first mention belongs to a character and this character has a relation that has the text of the second mention as a possible label (in our example, the character 'Richard' needs a relation that contains the label 'Ehefrau'). If this is the case, it can be determined that the second mention ('Ehefrau' in the example) belongs to the character which is the target of the relation. The cluster of the second mention is therefore merged into another cluster which has previously been identified to belong to the target character (ideally, there should only be one such cluster left by the time this sieve is applied).

Finally, we use the knowledge about relations for speculative merges of mentions that have a relation word as their text and are not in a cluster with any mention which is recognised as a name. If we find one of these mentions we go backwards in the text until we encounter a mention which belongs to a character that is the target of a relation with the relation word as a possible label (e.g. if we find a mention 'Ehefrau', encounter a mention which belongs to the character 'Lydia' and know that another character 'Richard' has a relation labelled 'Ehefrau' with 'Lydia' as the target, we merge both mentions' clusters).

# Results and Discussion

We evaluated our approach on six documents that were randomly picked from the documents of DROC [1] (Krug et al., 2018) for which we have summaries: *Die Hosen des Herrn von Bredow* by Willibald Alexis, *Stilpe* by Otto-Julius Bierbaum, *Der Stechlin* by Theodor Fontane, *Amerika* by Franz Kafka, *Anna Karenina* by Lev-Nikolaevic Tolstoj and *Uli der Pächter* by Jeremias Gotthelf. For each of these documents, two annotators, who were unfamiliar with the texts, separately created a document for userspecified knowledge by first reading the corresponding summary and then skimming the actual document. The time required for the creation of the meta data was between 5 to 15 minutes per file. Table 1 shows some characteristics of this user-specified knowledge: the number of characters, the total number of synonyms, the total number of relations and the total number of relation labels.

| Dokument | Characters | Synonyms | Relations | Labels |
|---|---|---|---|---|
| Alexis - Hosen (D) | 7 | 14 | 8 | 12 |
| Alexis - Hosen (M) | 7 | 21 | 3 | 5 |
| Bierbaum - Stilpe (D) | 4 | 7 | 0 | 0 |
| Bierbaum - Stilpe (M) | 3 | 4 | 1 | 3 |
| Fontane - Stechlin (D) | 9 | 10 | 2 | 4 |
| Fontane - Stechlin (M) | 8 | 12 | 3 | 3 |
| Kafka - Amerika (D) | 5 | 5 | 0 | 0 |
| Kafka - Amerika (M) | 4 | 7 | 0 | 0 |
| Tolstoj - Karenina (D) | 9 | 14 | 7 | 11 |
| Tolstoj - Karenina (M) | 5 | 8 | 5 | 9 |
| Gotthelf - Uli (D) | 4 | 6 | 2 | 3 |
| Gotthelf - Uli (M) | 3 | 4 | 0 | 0 |

*Tabelle 1: Characteristics of the user-specified knowledge created by the annotators: Number of characters, synonyms, relations and relation labels.*

The results of the algorithm with this userspecified knowledge are depicted in table 2 alongside the baseline results (the algorithm without any additional knowledge). We report the scores of the MUC metric (Vilain et al., 1995) which evaluates based on links between mentions and we report the results of the B-Cubed metric (Bagga and Baldwin, 1998) which evaluates based on cluster overlap between system entities and gold entities.

| | MUC | | | | $B^3$ | | | |
|---|---|---|---|---|---|---|---|---|
| Dokument | P | R | F1 | Δ | P | R | F1 | Δ |
| Alexis - Hosen | 84.89 | 71.74 | 77.76 | - | 74.87 | 29.01 | 41.81 | - |
| Alexis - Hosen (D) | 85.45 | 75.00 | 79.88 | +2.12 | 73.02 | 32.48 | 44.96 | +3.17 |
| Alexis - Hosen (M) | 84.33 | 73.1 | 78.31 | +0.55 | 74.00 | 39.96 | 51.90 | +10.09 |
| Bierbaum - Stilpe | 94.00 | 85.90 | 89.77 | - | 78.78 | 56.13 | 65.55 | - |
| Bierbaum - Stilpe (D) | 94.32 | 86.68 | 90.34 | +0.57 | 78.10 | 59.89 | 67.80 | +2.25 |
| Bierbaum - Stilpe (M) | 94.29 | 86.16 | 90.04 | +0.27 | 78.06 | 59.21 | 67.34 | +1.79 |
| Fontane - Stechlin | 92.29 | 83.44 | 87.64 | - | 83.24 | 45.79 | 59.08 | - |
| Fontane - Stechlin (D) | 92.33 | 83.88 | 87.90 | +0.26 | 81.12 | 55.84 | 66.15 | +7.07 |
| Fontane - Stechlin (M) | 92.40 | 84.75 | 88.41 | +0.77 | 79.14 | 62.31 | 69.72 | +10.64 |
| Kafka - Amerika | 94.75 | 88.11 | 91.31 | - | 85.17 | 65.59 | 74.11 | - |
| Kafka - Amerika (D) | 95.15 | 89.63 | 92.31 | +1.00 | 85.01 | 76.02 | 80.26 | +6.15 |
| Kafka - Amerika (M) | 95.11 | 89.02 | 91.97 | +0.66 | 85.12 | 70.76 | 77.27 | +3.16 |
| Tolstoj - Karenina | 84.73 | 79.49 | 82.03 | - | 64.94 | 51.00 | 57.13 | - |
| Tolstoj - Karenina (D) | 84.80 | 81.46 | 83.09 | +1.06 | 61.54 | 53.37 | 57.17 | +0.04 |
| Tolstoj - Karenina (M) | 84.57 | 80.06 | 82.25 | +0.22 | 62.35 | 51.76 | 56.56 | -0.57 |
| Gotthelf - Uli | 86.26 | 79.37 | 82.67 | - | 57.80 | 42.22 | 48.80 | - |
| Gotthelf - Uli (D) | 86.36 | 80.03 | 83.07 | +0.40 | 58.26 | 41.52 | 48.48 | -0.32 |
| Gotthelf - Uli (M) | 85.69 | 78.70 | 82.05 | -0.62 | 63.06 | 42.67 | 50.90 | +2.10 |
| Average Improvement (D) | +0.25 | +1.44 | +0.90 | - | -1.29 | +4.90 | +3.06 | - |
| Average Improvement (M) | -0.09 | +0.62 | +0.31 | - | -0.51 | +6.16 | +4.54 | - |
| Average Improvement (Avg) | +0.08 | +1.03 | +0.61 | - | -0.90 | +5.53 | +3.80 | - |

*Tabelle 2: Results of the rule-based algorithm without user-specified knowledge and with user-specified knowledge provided by two different annotators (D and M) on six randomly picked documents of DROC. The last three lines show the average improvement of the annotators.*

Depending on the text snippet, the improvements range from 0% up to almost 11% B-Cubed and up to 2% MUC score with an average improvement of about 4% B-Cubed. The small improvement of the MUC metric means, that with the help of the meta data, only relatively few links are improving, but these links reveal to be among the important ones, when the results of the B-Cubed metric is consulted. The improvements are mainly due to the improvements of the Recall, our algorithm is tuned to produce a conservative output and therefore does not attempt to merge references or entities that pose a high risk of failure. The usage of

meta data adds to the confidence for these merges and subsequently increases the Recall.

With user-specified knowledge at hand, we examined whether it just serves as an addition to our rule-based algorithm, or if it is able to replace the parts of our algorithm, which handle names and non-pronominal noun phrases (the algorithm cannot be replaced completely since user-specified knowledge does not help with pronouns). To assess this theory, we created the following algorithm: In the first step, all mentions which are identified to belong to the same character from user-specified knowledge (how this is done is described in the previous section) are merged into the same cluster. After that, only the parts of our algorithm which handle pronouns are applied. Table 3 shows the results of this string-matching baseline algorithm and the difference to the rule-based algorithm using the same user-specified knowledge. While the precision of the baseline is higher in all cases, its recall is lower, often by a rather large margin. This leads to the MUC score being slightly better in three cases but being worse in all other cases. The difference is even more noticable when looking at the B-Cubed scores: With two exceptions, they are always more than 5% worse than the results of the rule-based algorithm with user-specified knowledge. To summarize, one can say that the algorithm cannot be replaced by string-matching without a significant loss of quality.

| | MUC | | | | $B^3$ | | | |
|---|---|---|---|---|---|---|---|---|
| Dokument | P | R | F1 | Δ | P | R | F1 | Δ |
| Alexis - Hosen (D) | 92.41 | 61.96 | 74.18 | -5.70 | 89.05 | 18.59 | 30.76 | -14.20 |
| Alexis - Hosen (M) | 91.67 | 71.47 | 80.32 | +2.01 | 83.10 | 37.08 | 51.28 | -0.72 |
| Bierbaum - Stilpe (D) | 97.52 | 84.60 | 90.60 | +0.26 | 84.97 | 51.25 | 63.94 | -3.86 |
| Bierbaum - Stilpe (M) | 97.12 | 81.72 | 88.76 | -1.28 | 82.23 | 41.65 | 55.29 | -12.05 |
| Fontane - Stechlin (D) | 95.30 | 77.56 | 85.52 | -2.38 | 88.50 | 33.51 | 48.61 | -17.54 |
| Fontane - Stechlin (M) | 95.15 | 79.74 | 86.77 | -1.64 | 85.72 | 38.11 | 52.76 | -16.96 |
| Kafka - Amerika (D) | 96.15 | 81.10 | 87.99 | -4.32 | 89.50 | 40.93 | 56.17 | -24.09 |
| Kafka - Amerika (M) | 96.81 | 86.59 | 91.41 | -0.56 | 87.38 | 50.11 | 63.69 | -13.58 |
| Tolstoj - Karenina (D) | 89.22 | 75.28 | 81.66 | -1.43 | 74.88 | 38.43 | 50.79 | -6.38 |
| Tolstoj - Karenina (M) | 86.42 | 66.85 | 75.39 | -6.86 | 74.16 | 29.01 | 41.70 | -14.86 |
| Gotthelf - Uli (D) | 91.40 | 72.55 | 80.89 | -2.18 | 78.74 | 29.76 | 43.20 | -5.28 |
| Gotthelf - Uli (M) | 92.02 | 75.87 | 83.17 | +1.12 | 75.22 | 32.12 | 45.01 | -5.89 |

*Tabelle 3: Results of the String-Matching baseline using the user-specified knowledge created by the two annotators (D and M) and the difference to the results of the rule-based algorithm using the same knowledge.*

During the annotation of DROC, our experiments towards inter-annotator agreement revealed, that even human annotators only had an agreement of about 76% B-Cubed (Krug et al., 2018). Achieving a B-Cubed score of about 75% is therefore a milestone where the data seems reliable and we would expect the results to be usable for downstream tasks. An interesting aspect is that there is a large variance of improvement on different texts. Whenever relatively few named-entities that are communicating in a dialog are available in the text, the improvement is high (see *Amerika* or *Stechlin*) but the inverse effect occurs, when the author either is very vague with using names and aliases for characters or if there are many characters in the text in general. The quality of the results also depends on the summaries used. Using longer summaries (the ones we used were mostly rather short) or several different summaries per novel will likely lead to better results.

# Fußnoten

1. Note that in DROC, only persons are annotated. Coreference resolution usually also deals with other entities.

# Bibliography

**Aralikatte, R., Lent, H. / Gonzalez, A. V. / Hershcovich, D. / Qiu, C. / Sandholm, A. / Ringaard, M. / Søgaard, A.** (2019). Rewarding coreference resolvers for being consistent with world knowledge. *arXiv preprint arXiv:1909.02392*.

**Bagga, A. / Baldwin, B.** (1998): Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–566. Granada.

**Bryl, V. / Giuliano, C. / Serafini, L. / Tymoshenko, K.** (2010): Using background knowledge to support coreference resolution. In *ECAI*, volume 10, pages 759–764. Citeseer.

**Joshi, M. / Chen, D. / Liu, Y. / Weld, D. S. / Zettlemoyer, L. / Levy, O.** (2019): Spanbert: Improving pre-training by representing and predicting spans. *arXiv preprint arXiv:1907.10529*.

**Krug, M. / Puppe, F. / Jannidis, F. / Macharowsky, L. / Reger, I. / Weimar, L.** (2015): Rule-based coreference resolution in german historic novels. In *Proceedings of the Fourth Workshop on Computational Linguistics for Literature*, pages 98–104.

**Krug, M. / Puppe, F. / Reger, I. / Weimer, L. / Macharowsky, L. / Feldhaus, S. / Jannidis, F.** (2018): Description of a corpus of character references in german novels - DROC [Deutsches ROman Corpus]. In *DARIAH-DE Working Papers*. DARIAH-DE.

**Lee, K. / He, L. / Lewis, M. / Zettlemoyer, L.** (2017): End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197.

**Lee, K. / He, L. / Zettlemoyer, L.** (2018): Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692.

**Ng, V.** (2007): Shallow semantics for coreference resolution. In *IJcAI*, volume 2007, pages 1689–1694.

**Rahman, A. / Ng, V.** (2011): Coreference resolution with world knowledge. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language TechnologiesVolume 1*, pages 814–824. Association for Computational Linguistics.

**Richardson, M. / Domingos, P.** (2006): Markov logic networks. *Machine learning*, 62(1-2):107–136.

**Suchanek, F. M. / Kasneci, G. / Weikum, G.** (2007): Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706. ACM.

**Vilain, M. / Burger, J. / Aberdeen, J. / Connolly, D. / Hirschman, L.** (1995): A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*, pages 45–52. Association for Computational Linguistics.