Resources and Evaluations for Danish Entity Resolution

Maria Barrett† **Hieu Trong Lam**† **Martin Wu**†
Ophélie Lacroix◇ **Barbara Plank**† **Anders Søgaard** ○

† Computer Science Department, IT University of Copenhagen, Denmark
[mbarrett,trol,mawu,bapl]@itu.dk
◇ Alexandra Institute, Denmark, ophelie.lacroix@alexandra.dk
○ Department of Computer Science, University of Copenhagen, Denmark
soegaard@di.ku.dk

Abstract

Automatic coreference resolution is understudied in Danish even though most of the Danish Dependency Treebank (Buch-Kromann, 2003) is annotated with coreference relations. This paper describes a conversion of its partial, yet well-documented, coreference relations into coreference clusters and the training and evaluation of coreference models on this data. To the best of our knowledge, these are the first publicly available neural coreference models for Danish. We also present a new entity linking annotation on the dataset using WikiData identifiers, a named entity disambiguation (NED) dataset, and a larger automatically created NED dataset enabling wikily supervised NED models. The entity linking annotation is benchmarked using a state-of-the-art neural entity disambiguation model.

1 Introduction

The Danish Dependency Treebank (DDT) (Buch-Kromann, 2003) is a beneficial resource for Danish NLP that contains several annotation layers. Most of the layers were annotated as part of the Copenhagen Dependency Treebank project, but a conversion of the dependency syntax annotation into Universal dependencies (Johannsen et al., 2015; Nivre et al., 2020) and the addition of named entities annotation layers (Hvingelby et al., 2020; Plank, 2019; Plank et al., 2020) are newer additions. The partial coreference annotation has received no attention for NLP purposes despite being very well documented. This paper describes converting the coreference relations into coreference clusters and a new entity linking annotation with unique Wikidata item identification codes (QIDs) (Vrandečić, 2012) on the same data.

Entity linking is the task of detecting mentions and matching the mentions to a knowledge base. The two annotation layers—coreference and entity linking—complement each other as two types of entity resolution. In practice, entity linking can be reduced to a binary classification task, called named entity disambiguation (NED), thereby simplifying the task and omitting mention detection.

The coreference-annotated data is benchmarked using a strong neural model. The entity linking annotation is turned into a NED dataset and benchmarked using a state-of-the-art NED model. We automatically create a larger NED train set from Wikipedia articles for train set augmentation, and we observe an improvement for the wikily supervised models. All data and models are publicly available.[1]

2 Related work

Danish anaphors have received some attention: Navarretta (2000) shows that Danish deictics are used in more contexts than the English ones and Houser et al. (2006) specifically discuss the use of verb phrase pronominalization in Danish. Danish has gendered possessive pronouns, but non-gendered reflexive pronouns. This has made it useful as an unambiguous testbed for gender bias in natural language inference models, machine translation models, and language models (González et al., 2020). But automatic coreference resolution for Danish has received no attention, and there was no established evaluation set for this task.

Linked resources such as Wikipedia enable multi-lingual entity linking/NED models and datasets, and Danish is often among the evaluation languages (Pan et al., 2017; McNamee et al., 2011). DBpedia Spotlight[2] is the most recent entity linking system that also supports Danish. But due to this being a different task from NED, we can not compare our model to DBpedia Spotlight.

[1] https://github.com/alexandrainst/danlp
[2] https://www.dbpedia-spotlight.org/

Proceedings of the 4th Workshop on Computational Models of Reference, Anaphora and Coreference (CRAC 2021), pages 63–69
Punta Cana, Dominican Republic, November 10–11, 2021.

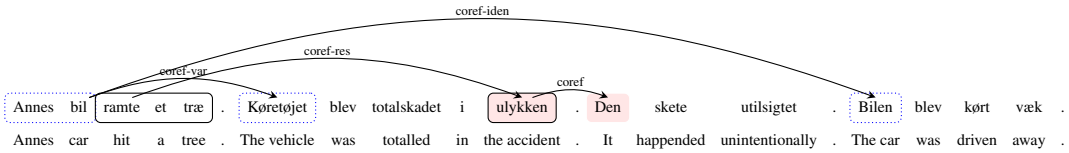

Figure 1: A constructed example of a coreference-annotated paragraph with binary relations as graphs and three clusters marked with background colour or borders around the word spans. The English word-by-word translation is below. Translation: "Anne's car crashed into a tree. The vehicle was damaged beyond repair in the accident. It happened unintentionally. The car was towed."

	N	LA	UA	REL
COREF	238	61	64	89
COREF-VAR	146	59	72	73
REF	70	90	91	94
COREF-IDEN	62	73	81	77
COREF-RES	36	62	67	75
COREF-EVOL	1	0	100	0

Table 1: Labelled agreement (LA), Unlabelled agreement (UA), only relation name (REL) and support (N) for the doubly annotated subset. Numbers are taken directly from the DDT documentation.[3]

3 Dataset

The coreference-annotated part spans 2/3 of the DDT and is documented in Korzen and Buch-Kromann (2011). This subset encompasses 341 documents/3,403 sentences/64,076 tokens. The source is the PAROLE Corpus (Bilgram and Keson, 1998). The domain is mostly newswire (299 documents) but also a small fraction of magazine text (41 documents) and news broadcast (1 document). We refer to this dataset as DACOREF.

4 Coreference

Annotation and conversion The DDT documentation[3] is extensive and this section's details concerning the annotation come from this resource. We refer to the DDT documentation for more details.

Key observations about the annotation scheme: Like the Ontonotes (Weischedel et al., 2013) coreference annotation, singletons are not annotated. The annotation does not label attributive noun phrases connected through copula verbs such as "to be" ("at være"). Verb phrases can only be linked if they are coreferent with a noun phrase.

The dataset is annotated with binary relations between core nodes. We consider nine different coreference labels that we merge heuristically into clusters. We omit the associative anaphors. Coreference clusters are sets of text spans that all refer to the same entity and labels are omitted. Each cluster is uniquely numbered.

We merge the following relations if they span the same target words: COREF (coreferential personal pronoun), COREF-ELL (elliptic anaphor demonstrative pronoun), COREF-EVOL (evolving anaphor), COREF-IDEN (coreferential noun phrase with complete lexical identity), COREF-IDEN.SB (coreferential noun phrase with lexical identity in the noun but lexical variety in some other (typically attributive) component), COREF-VAR (coreferential noun phrase with lexical variety in the noun), and REF (syntactically determined coreference, e.g., relative pronouns).

COREF-RES (resumptive anaphor clause or predicate) and COREF-RES.PRG (pragmatic resumptive anaphor) are also used for clustering but not merged with any other label, nor each other. Figure 1 shows a constructed example where COREF-VAR, COREF-IDEN and COREF are merged when on the same target word(s), but COREF-RES is not, thereby making `ulykken` part of two clusters.

Seven documents are doubly annotated, and they are assumed to form the basis of the reported annotation quality in the documentation, which we reprint in Table 1. COREF-ELL, COREF-IDEN.SB, and COREF-RES.PRG are subclasses of COREF, COREF-IDEN and COREF-RES, respectively. Separate scores were not reported for those but they are assumed to be included in the parent class score. DACOREF use the annotations of one annotator, "Lotte".

The coreference annotation only contains a label on the core node. For DACOREF, we propagate the label to the relevant span. Due to several instances of the label not being on the core node, this is done manually by one annotator. The spans encompass the relevant noun phrases. When linking

[3] https://github.com/mbkromann/copenhagen-dependency-treebank/blob/master/manual/cdt-manual.pdf

verb phrases, the span covers the entire verb phrase. This is different from Ontonotes (Weischedel et al., 2013), where the span only covers the head verb.

Even though the DDT has been split into train, development and test sets as part of the Universal Dependencies (UD) conversion (Johannsen et al., 2015), a new split is created for document-level coreference resolution since the UD dataset scrambled the sentences across documents. The train split contains 290 documents, the development set 23, and the test set 28 documents.

Coreference models We provide benchmark scores for two strong neural models using three different transformer representations. Lee et al. (2017) (LEE2017) presented the first neural end-to-end coreference model in which the spans are learned in the same training pass as the pairwise clustering. The pairwise clustering may produce globally inconsistent clusters, and Lee et al. (2018) (LEE2018) presented a higher-order upgrade that did not only consider pairs of spans but a matrix of all spans to counter global inconsistencies. Joshi et al. (2019) showed that the model from Lee et al. (2018) performed even better on English data when using transformer-based models instead of the word embeddings of the original implementations. We follow their approach and try representations from three different pre-trained models: Danish BERT (DABERT)[4] and two multilingual, cased models: multilingual BERT (MBERT) (Devlin et al., 2019) and the base model of XLM-Roberta (XLM-R) (Conneau et al., 2020). Instead of the TensorFlow implementations released by Joshi et al. (2019), we use the PyTorch-based implementation from AllenNLP 1.3.0.[5] with PyTorch version 1.7.1. A description of the tuning process is in Appendix A. After model selection, we retrain the models for a maximum of 1200 epochs with early stopping and a patience of 10.

Coreference results The coreference benchmark results are presented in Table 2. Models based on the two multi-lingual, cased transformer models perform a lot better than the uncased DABERT. The best model is LEE2018 trained with XLM-R.

	MODEL	F1	P	R	MR
DABERT	LEE2017	0.477	0.587	0.402	0.729
	LEE2018	0.313	0.655	0.207	0.683
MBERT	LEE2017	0.630	0.679	0.587	0.870
	LEE2018	0.532	0.625	0.463	0.854
XLM-R	LEE2017	0.623	0.668	0.585	0.822
	LEE2018	**0.640**	**0.699**	**0.592**	**0.880**

Table 2: Coreference results: M(ention) R(ecall), average P(recision), R(ecall) and F1 across MUC, B^3, and CEAFe. The best result per column is boldfaced.

5 Entity linking

Entity linking annotation The resource was annotated with QIDs in the spring of 2020 with the WikiData entries available at that time in a semi-manual process: First, all tokens in named entities were used to search the MediaWiki API. Given the entire list of WikiData matches for each token, one annotator decided which QID match was correct for each. We did not search for entire mentions since mentions were not always equal to the WikiData label and thus returned too many wrong QID suggestions leaving this method inefficient. In the case of multi-word entity names, we searched for possible QID matches for each token and manually selected the correct one. One annotator decided in each case whether "Margrethe" referred to the Danish queen, a foundation in her name or another person/ship etc. This was checked by another annotator, who also manually added the QID to the correct span in the text. Both were native speakers of Danish. The average accuracy of the set of QIDs per document of annotator 1 compared to annotator 2 is 0.98 (std. 0.05). Furthermore, the second annotation process also included adding a generic QID for words that matched one of 47 categories (Ship Q11446, Award Q618779, Automobile model Q3231690, Hospital Q16917 etc.) for which no specific Wikidata entry existed. The list of all generic QIDs can be seen in Appendix B. In total, 7,173 tokens were annotated with a QID. 2,193 unique QIDs were used.

Construction of the DANED dataset Entity disambiguation is a binary classification variant of entity linking. Given a sentence, the entity name as printed in the sentence, and a QID; the model classifies whether this QID is the mentioned entity in the sentence. The task requires creating a new classification dataset also containing negative examples. Only sentences from DACOREF that have at least one QID annotation are part of the datasets.

We create the negative examples in the following manner: For each token annotated with a QID in the train, development, and test set, we search

[4] https://github.com/botxo/nordic_bert
[5] https://github.com/allenai/allennlp

SENTENCE	ENTITY	QID	KG CONTEXT	LABEL
The same sentence could be heard when Elvis had left the scene after a Las Vegas show.	Elvis	Q303	birth name Elvis Aaron Presley given name Elvis given name Aaron country of citizenship USA place of birth Tupelo place of death Graceland spouse Priscilla Presley child Lisa Marie Presley place of burial Graceland Commons category Elvis Presley LCAuth n78079487 VIAF 23404836 GND identificator 118596357 ISNI 0000 0001 2124 1960 ISNI 0000 0003 6863 8871 occupation film actor occupation rock singer occupation screen writer occupation guitarist occupation soldier occupation pianist occupation actor	1
The same sentence could be heard when Elvis had left the scene after a Las Vegas show.	Elvis	Q5368160	Commons category Firefighting helicopters in Australia image N179AC-Elvis-739.jpg instance of aircraft Freebase-ID /m/0271gwk	0

Table 3: Example of the representation of two samples for the NED model. All input to the model is in Danish; only this example is shown in English.

WikiData using SPARQL for other matches than the correct QID but with the same label name. If possible, up to two negative examples per QID are included. The same sentence occurs several times in each split due to negative examples and possibly multiple QIDs in the same sentence, but the same sentence does not occur across splits.

We used the MediaWiki API to obtain all WikiData knowledge graph (KG) contexts in Danish for both the positive and negative examples. Due to model constraints, we cut off the properties after 512 characters. An example of the representation of an instance is presented in Table 3.

The train-development-test split is determined by the DACOREF splits and ended in a 78-9-13% distribution. Class balance and split size after the construction of negative samples and dataset size are shown in Figure 2. This dataset is referred to as DANED.

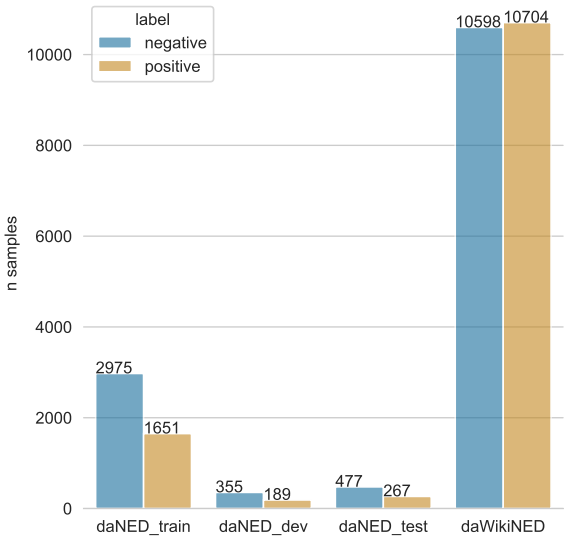

Figure 2: Distribution of class labels and data set sizes in the datasets for evaluating and training NED models.

Train set augmentation We automatically construct a large Danish NED dataset from Wikipedia, DAWIKINED. We use a list of 391,102 Danish articles names from the latest Wiki dump[6]. Initially, we further gather 82,150 person names that have Danish articles from the SPARQL endpoint. We fetch the Wikipedia text using the SPARQL API for each of the article names and split it into sentences using the NLTK library. We then pick the longest sentence that also contains the article title[7]. If possible, we fetch the corresponding QID, the corresponding KG context and the description for this entity and add it to the dataset. We fetch up to two negative examples per QID. Final DAWIKINED dataset size is in Figure 2.

NED model The model by Mulang' et al. (2020) takes the best model from Yang et al. (2019) called dynamic context augmentation with supervised learning (DCA-SL) but fine-tunes transformer model representation during training. The model leverages KG contexts from a knowledge base during training and evaluation. Mulang' et al. (2020) reported state-of-the-art performance on the AIDA-CONLL dataset (Hoffart et al., 2011). During training, we fine-tune either DABERT and XLM-R representations. In the code, we modify

	TRAIN SET	P	R	F1
XLM-R	DANED	0.76	0.90	0.82
	DAWIKINED	0.64	0.62	0.63
	BOTH	**0.84**	0.85	0.85
DABERT	DANED	0.8	**0.91**	0.85
	DAWIKINED	0.82	0.54	0.65
	BOTH	**0.84**	0.88	**0.86**

Table 4: NED results on the DANED test set. The best result per column is boldfaced.

[6]https://dumps.wikimedia.org/dawiki/

[7]We use fuzzy string matching with a similarity threshold of 85 to match the article name in the sentence because some of the article names are slightly different when appearing in the text, especially personal names.

that the KG context was not scrambled and reduced to the vocabulary using the `set()` function but keep the rest as-is. The tuning process is described in Appendix C.

NED results Table 4 presents the results of the NED evaluation. DABERT is slightly better than XML-R. We observe that the augmented training with DAWIKINED seems to help both models. The best model builds on DABERT and trains on the DANED train set and DAWIKINED (BOTH).

6 Conclusion

We have presented a semi-manual conversion of coreference relations into coreference clusters and a novel entity linking annotation for Danish. The latter annotation is transformed into a named entity disambiguation dataset, and we further described the automatic construction of a Danish Wikipedia named entity disambiguation dataset. We have reported the first benchmarks for Danish coreference resolution and an evaluation of our named entity disambiguation dataset with train set augmentation.

Acknowledgements

Maria Barrett is supported by a research grant (34437) from VILLUM FONDEN. Barbara Plank is in part supported by the Independent Research Fund Denmark (DFF) grant 9131-00019B. The Alexandra Institute is supported by the performance contract "*Digital sikkerhed, tillid og dataetik*" funded by the Danish Ministry of Higher Education and Science.

7 Bibliographical References

References

Thomas Bilgram and Britt Keson. 1998. The construction of a tagged Danish corpus. In *Proceedings of the 11th Nordic Conference of Computational Linguistics (NODALIDA 1998)*, pages 129–139, Copenhagen, Denmark. Center for Sprogteknologi, University of Copenhagen.

Matthias Buch-Kromann. 2003. The Danish Dependency Treebank and the DTAG treebank tool. In *2nd Workshop on Treebanks and Linguistic Theories (TLT)*, pages 217–220, Växjö, Sweden. Växjö University Press.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ana Valeria González, Maria Barrett, Rasmus Hvingelby, Kellie Webster, and Anders Søgaard. 2020. Type B reflexivization as an unambiguous testbed for multilingual multi-task gender bias. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2637–2648, Online. Association for Computational Linguistics.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 782–792, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Michael J Houser, Line Mikkelsen, Maziar Toosarvandani, Erin Brainbridge, and Brian Agbayani. 2006. Verb phrase pronominalization in Danish: Deep or surface anaphora? In *Proceedings of the thirty-fourth Western Conference On Linguistics (WECOL)*, volume 17, Fresno, CA, US. California State University.

Rasmus Hvingelby, Amalie Brogaard Pauli, Maria Barrett, Christina Rosted, Lasse Malm Lidegaard, and Anders Søgaard. 2020. DaNE: A named entity resource for Danish. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4597–4604, Marseille, France. European Language Resources Association.

Anders Johannsen, Héctor Martínez Alonso, and Barbara Plank. 2015. Universal dependencies for Danish. In *International Workshop on Treebanks and Linguistic Theories (TLT14)*, pages 157–167, Warsaw, Poland. Institute of Computer Science, Polish Academy of Sciences.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. 2019. BERT for coreference resolution: Baselines and analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5803–5808, Hong Kong, China. Association for Computational Linguistics.

Iørn Korzen and Matthias Buch-Kromann. 2011. Anaphoric relations in the Copenhagen Dependency Treebanks. In *Proceedings of DGfS Workshop – Göttingen, Germany*, pages 83–98, Bochum, Germany. Ruhr-Universität Bochum, Sprachwissenschaftliches Institut.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 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, New Orleans, Louisiana. Association for Computational Linguistics.

Paul McNamee, James Mayfield, Dawn Lawrie, Douglas W. Oard, and David Doermann. 2011. Cross-language entity linking. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 255–263, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Isaiah Onando Mulang', Kuldeep Singh, Chaitali Prabhu, Abhishek Nadgeri, Johannes Hoffart, and Jens Lehmann. 2020. Evaluating the impact of knowledge graph context on entity disambiguation models. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 2157–2160, Virtual event, Ireland. Association for Computing Machinery, NY, US.

Costanza Navarretta. 2000. Abstract anaphora resolution in Danish. In *1st SIGdial Workshop on Discourse and Dialogue*, pages 56–65, Hong Kong, China. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Barbara Plank. 2019. Neural cross-lingual transfer and limited annotated data for named entity recognition in Danish. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 370–375, Turku, Finland. Linköping University Electronic Press.

Barbara Plank, Kristian Nørgaard Jensen, and Rob van der Goot. 2020. DaN+: Danish nested named entities and lexical normalization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6649–6662, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Denny Vrandečić. 2012. Wikidata: A new platform for collaborative data collection. In *Proceedings of the 21st international conference on world wide web*, pages 1063–1064, New York, NY, USA. Association for Computing Machinery.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston. 2013. OntoNotes release 5.0.

Fan Yang, Xiaochang Peng, Gargi Ghosh, Reshef Shilon, Hao Ma, Eider Moore, and Goran Predovic. 2019. Exploring deep multimodal fusion of text and photo for hate speech classification. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 11–18, Florence, Italy. Association for Computational Linguistics.

A Tuning of coreference models

We tune the task learning rate (LR) and the transformer LR. Models are tuned individually using full grid search with early stopping for 50 epochs and a patience of 10 epochs. Tuned models are trained on the DACOREF train set, and the best settings are selected based on the DACOREF development set. Tuning the values with one step on the logarithmic scale hurt the model, so we tuned on smaller steps as outlined in Table 5.

The remaining parameters were kept fixed according to the original parameters. The development set was used both for early stopping and to select the best model. The criterion for model selection was the average F1 score over MUC, CEAF, and B3 F1 scores.

MODEL	TASK LR	TRANSFORMER LR
LEE2017	{9e−02, 1e−03, 2e−03}	{1e−05, 2e−05, 3e−05}
LEE2018	{2e−04, 3e−04, 4e−04}	{1e−05, 2e−05, 3e−05}

Table 5: Grid search parameters for tuning coreference models

B Generic QID's in entity linking annotation

Family name Q101352
Unisex nickname Q49614
Female given name Q11879590
Male given name Q12308941
Unisex name Q3409032
Artist name Q483501
Magazine Q41298
Hotel Q27686
Work of art Q838948
Governmental administrative unit in Denmark Q21268738
Municipal Police Q1758690
Road Q34442
Cohousing Q1107167
Postal address Q319608
Museum Q33506
Security (tradeable financial asset) Q169489
Geographic location Q2221906
Radio program Q1555508
Tv program Q15416
Product/goods Q2424752
Department within organisation Q2366457
Organization Q43229
Sports venue Q1076486
Dish Q746549 (only one instance)
Event Q1656682
Fleet Q189524
University Q3918
Disease Q12136
Coast Q93352
Ship Q11446
Award Q618779
Automobile model Q3231690
Project (also Inquiry) Q170584
Hospital Q16917
Amusement ride Q1144661
Sports team Q12973014
Building Q41176
Bill (proposed law) Q686822
Restaurant Q11707
People/ethnic group Q2472587
Educational institution Q2385804
Shop Q213441
Publication Q732577
Legislation Q49371
Night club Q622425
Newspaper Q11032
Prison Q40357
Army Q37726.

C Tuning of NED models

For training, we use the default settings from Mulang' et al. (2020) apart from the following values: train batch size, number of training epochs, LR, and warm-up ratio. Values are trained on the DANED train set, and the best models are selected based on the development set F1 score.

We tuned models using both pre-trained representations individually without any train set augmentation. The optimal hyperparameters are subsequently used for all models with this transformer architecture. During tuning, the models are trained with the following hyperparameters apart from the default settings: batch size of 8, for 1 epoch, the learning rate set to 4e-5, and the warm-up ratio set to 0.06. For hyperparameter tuning, we change one of the hyper-parameters from the default hyperparameter setting. Below are the tried values. The best value, which accidentally is the same for both transformer representations, is marked with bold:

- train batch size: **8**, 16
- num train epochs: 4, **8**
- learning rate: **4e-05**, 4e-06
- warmup ratio: **0.07**, 0.08, 0.1