DaNLP: An open-source toolkit for Danish Natural Language Processing

Amalie Brogaard Pauli[♦] **Maria Barrett**[♠] **Ophélie Lacroix**[♦] **Rasmus Hvingelby**[♣*]
[♦]The Alexandra Institute {amalie.pauli, ophelie.lacroix}@alexandra.dk
[♠] The IT University of Copenhagen mbarrett@itu.dk
[♣] Fraunhofer IIS rasmus.hvingelby@iis.fraunhofer.de

Abstract

We present an open-source toolkit for Danish natural language processing (NLP), enabling easy access to Danish NLP's latest advancements. The toolkit features wrapper functions for loading models and datasets in a unified way using third-party NLP frameworks. The toolkit is developed to enhance community building, understanding the need from industry and knowledge sharing. As an example of this, we present Angry Tweets: An Annotation Game to increase Danish NLP awareness and create a new sentiment-annotated dataset.

1 Introduction

Danish is the official language in Denmark. It is mainly spoken by the approximately 6M people in Denmark[1]. In natural language processing (NLP), Danish is considered a medium-resource language (Joshi et al., 2020). There is, however, limited availability of Danish models and tools (Kirkedal et al., 2019). We believe to increase the availability of NLP resources, we need to engage academia and industry to leverage synergy effects.

In this demonstration paper, we present an open-source, Danish NLP toolkit: *DaNLP*. It contains trained models for named entity recognition (NER), part-of-speech (PoS) tagging, sentiment analysis, parsing, coreference resolution as well as word embeddings and datasets. It is developed in close collaboration with industry and academic partners and aims at strengthening knowledge sharing and community building.

The toolkit makes recent advances in Danish NLP more available and applicable. It is a single entry for accessing Danish NLP resources through a consistent interface. The toolkit consists of resources developed by others and new models and datasets developed within the project guided by what is presently relevant for industry. In the same spirit, we ensure industry-friendly licences, i.e., the resources are licensed for commercial use, ideally without copyleft restrictions. The toolkit employs a unified syntax for loading and applying models inspired by frameworks like scikit-learn (Buitinck et al., 2013) and spaCy (Honnibal et al., 2020).

The overall scope of the DaNLP project is to engage a community of professionals from academia and industry around Danish NLP. As a way of showcasing what is needed concerning annotation and to engage people in the development of Danish NLP tools, a crowdsourcing game is launched as part of the project. This paper's main contribution is to demonstrate a resource enabling industry's adoption of NLP for a medium-resource language.

2 Related Work

There are several NLP tools for Danish which we will not review here, but extensive lists exist such as the one by Finn Årup Nielsen.[2]

We consider an NLP toolkit to be a collection of resources that spans several NLP tasks in one unified framework. This section provides a brief overview of NLP toolkits for Danish and a non-exhaustive selection of comparable languages.

Derczynski et al. (2014) presented an open-source information extraction toolkit for Danish supporting tokenization, named entity recognition (NER) and part-of-speech (PoS) tagging. How-

[*]Rasmus Hvingelby carried out this work while affiliated with the Alexandra Institute.

[1]"Danish" refers to standard Danish. The minority languages and dialects of Denmark are not within the scope of this project.

[2]http://www2.imm.dtu.dk/pubdb/edoc/imm6956.pdf

ever, they are released with a copyleft or non-commercial licence, making them less appealing for industry.

Several multilingual toolkits have some support for Danish, e.g., the Natural Language ToolKit (Loper and Bird, 2002), Polyglot (Al-Rfou et al., 2013), SpaCy (Honnibal et al., 2020), Stanza (Qi et al., 2020), UDPipe (Straka et al., 2016), and Apache OpenNLP (Apache Software Foundation, 2014).

For other medium-resource Germanic languages, language-specific toolkits exist, e.g., Icelandic (Þorsteinsson et al., 2019; Loftsson and Rögnvaldsson, 2007) and Dutch (Bosch et al., 2007; Bouma et al., 2001).

3 The DaNLP toolkit

The DaNLP toolkit contains wrapper functions utilising well-maintained third-party NLP frameworks such as spaCy (Honnibal et al., 2020), Flair (Akbik et al., 2018), Gensim (Řehůřek and Sojka, 2010) and Transformers (Wolf et al., 2020).

The documentation[3] for the toolkit provides an overview of the resources with credits to developers, benchmark results, training details, and code snippets for loading and using the models and datasets.

The resources available through the toolkit include both resources developed by others and resources developed specifically as part of the DaNLP project. Therefore, in the following subsections, a † indicates that a resource was created/trained/annotated as part of the DaNLP project. In the opposite case, a reference is supplied.

3.1 Datasets

This subsection provides an overview of available datasets through the DaNLP toolkit.

The Danish Dependency Treebank (DDT) (Buch-Kromann, 2003) consists of texts from the Danish PAROLE corpus (Keson, 1998). The treebank has several layers of annotations but those currently relevant for the models in the toolkit are the *Universal Dependency*(UD) conversion (Johannsen et al., 2015) and the *coreference* annotation. The treebank was additionally annotated with *named entities* and released as the DANE dataset† (Hvingelby et al., 2020).

NER Besides the DDT annotation (described above), the toolkit also supports the Danish part of WikiANN (Pan et al., 2017) containing Wikipedia articles.

Sentiment Different small sentiment datasets are included: The lcc-sentiment[4] which contains manual annotated sentences from the Leipzig Corpora Collection (Quasthoff et al., 2006), europarl-da-sentiment[5], Europarl Sentiment2†, and Twitter Sentiment† described in §4.

Word similarity For evaluating word representations, DaNLP includes two-word similarity datasets: the Danish Similarity Dataset (Schneidermann et al., 2020) and WordSim-353 (Finkelstein et al., 2001).

DanNet (Pedersen et al., 2009), a Danish WordNet (lexical database), is implemented in DaNLP with functions for finding synonyms, hypernyms, etc.

3.2 Models

This section provides an overview of the best performing models[6] integrated into the toolkit.

NER The best NER model† is fine-tuned on a Danish pre-trained BERT model (Devlin et al., 2019)[7] and benchmarked on the DaNE annotation† (Hvingelby et al., 2020) using the Transformers architecture from Huggingface (Wolf et al., 2020).

PoS-tagging The best PoS model† implemented in the toolkit is trained using the Flair framework. It is trained and tested on the Danish UD treebank (Johannsen et al., 2015).

Sentiment The toolkit includes sentiment models for three-way polarity†, subjectivity-objectivity detection†, and eight-way emotion detection†. The best polarity and subjectivity-objectivity detection models are trained and benchmarked on Twitter Sentiment† and Europarl Sentiment2† by fine-tuning the Danish BERT model. The polarity models are additionally

[3] https://danlp-alexandra.readthedocs.io

[4] https://github.com/fnielsen/lcc-sentiment

[5] https://github.com/fnielsen/europarl-da-sentiment

[6] based on the most common evaluation metric for the task.

[7] https://github.com/botxo/nordic_bert

benchmarked on lcc-sentiment and europarl-da-sentiment. The Emotion detection model is trained on social media data by fine-tuning the Danish BERT model; however, it was impossible to open-source the data, see §5.

Coreference Resolution The best coreference model[†] is the AllenNLP (Gardner et al., 2018) implementation of Lee et al. (2018) fine-tuned using XLM-Roberta (Conneau et al., 2019) instead of static word embeddings, in line with Joshi et al. (2019). Models are benchmarked on the DDT (Buch-Kromann, 2003).

Dependency Parsing and Chunking We support dependency parsing through the spaCy framework using a model[†] trained on the DDT dataset. We also provide wrapper-code for deducing noun-phrase chunks from predicted dependency trees.

3.3 Text representation

The toolkit contains static word embeddings pre-trained by third-parties[8] with word2vec (Bojanowski et al., 2017) and fastText (Mikolov et al., 2013). Dynamic word embeddings[†], trained using the Flair architecture (Akbik et al., 2018) are also available in the toolkit, as well as embeddings derived from the Danish BERT language model.[9]

3.4 DaNLP: Selected examples of usage

The goal of the DaNLP project is to make datasets and models easily accessible through a unified syntax. Therefore, the package provides consistent functions for loading datasets through prominent frameworks such as spaCy or Flair – e.g., for training purposes – or in standard datatypes or formats such as DataFrames[10] or CoNLL-U[11]. Below is an example of several possibilities for loading the DDT:

```
#Danish Dependency Treebank
from danlp.datasets import DDT
ddt = DDT()
spacy_corpus= ddt.load_with_spacy()
flair_corpus = ddt.load_with_flair()
conllu_format = ddt.load_as_conllu()
```

Models can also be loaded with a unified syntax. However, there are differences in applying them based on the framework they are trained with, though most of them are provided with simple prediction functions that take a sentence as input. Below is an example of how to load and use the Emotion detection model:

```
# Emotion Detection
from danlp.models import (
   load_bert_emotion_model )
clf = load_bert_emotion_model()
clf.predict("Jeg ser frem til det")
```

4 Angry Tweets: An Annotation Game

To advance the field of Danish NLP, there is a need for task-specific annotated corpora for training and benchmarking models. (Kirkedal et al., 2019; Sprognævn, 2019). The DaNLP project has previously annotated a corpus using a traditional approach, i.e., with a few trained annotators. However, such annotations are expensive and time-consuming. Therefore, we propose collaborative crowdsourcing, designed as a game. A similar approach is previously seen in Öhman et al. (2018). Like their gamified emotion annotation setup, we also asked participants to annotate a few gold-annotated sentences as well as sentences previously annotated by other crowd annotators in order to assess the annotation. The main motivation, besides creating a new Twitter sentiment corpus for the toolkit, was engaging professionals and other people interested in the development of Danish NLP and communicating what is needed in terms of annotation work. The game was, therefore, announced through social media, a blog post, and the Danish medium Data Tech.[12] The hope was to motivate volunteer participants to contribute to the development of Danish NLP. The gamification element (failing is an option, and there was also a possibility to win a symbolic prize) is meant to peak people's interest and motivate them to supply high-quality annotations. We made an effort to keep a light and fun tone with a storyline including a swan, the project logo.

The game interface The game consists of eight rounds with four tweets per round. Figure 1 shows

[8] https://loar.kb.dk/handle/1902/329, https://fasttext.cc/docs/en/crawl-vectors.html, https://github.com/danish-stance-detectors/RumourResolution

[9] https://github.com/botxo/nordic_bert

[10] https://pandas.pydata.org/

[11] https://universaldependencies.org/format.html

[12] https://pro.ing.dk/datatech/article/angry-tweets-vaer-med-til-bygge-datasaet-over-foelelsesladede-tweets-9496

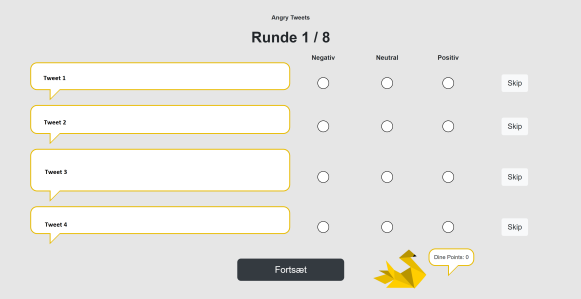

Figure 1: One annotation page in Angry Tweets.

one round. The tweets are annotated with three-class sentiment (positive, neutral, negative). As a part of a defensive task design (Sabou et al., 2014), participants were on every second page asked to annotate one gold-annotated tweet, and on each round, the completion time was measured. Not passing any of these checks triggered game over. In each round, participants annotated one sentence already annotated by another annotator and was rewarded with a point if their annotation matched the previous annotation.

Statistics The game was completed 114 times. 82% of players completing the game submitted a contact email to participate in the competition for a prize, indicating that some participants were not motivated by the prize. The tweets are collected through a list of commonly used Danish hashtags and posted between January and May 2019. The corpus consists of 4921 annotated tweets, where 1266 is double annotated with an inter-annotator agreement of 65%. The majority is annotated through the game, but 1727 was annotated by one trained annotator.

5 Knowledge In Knowledge Out

The development of DaNLP is industry-focused. Therefore, the DaNLP team is in dialogue with Danish companies and government agencies to understand their needs. The project also shares knowledge and disseminates.

Throughout the project, there have been dialogues with around 35 companies consisting of both start-ups and larger tech companies, as well as eight different government agencies. There is a large spread in the maturity of using and understanding NLP across organisations. Some companies are pushing the field of Danish NLP forward, and their requests are generally more data; large, raw text corpora and annotated corpora. Other companies are new to the field and mostly driven by curiosity, and a third category consists of companies with a more task-oriented desire. Here, we especially noted a need for better performing models for NER and sentiment analysis. Therefore, these tasks were the initial focus of the toolkit. To stay in close contact with industry, two collaborations with companies were constituted: One with a media monitoring company, Infomedia A/S, to improve their existing NER system for news articles. The other collaboration was with the Danish Broadcasting Corporation to monitor the mood on their social media platform.

The knowledge-sharing part is aimed at making more people and companies aware of the possibilities of NLP. Therefore the project includes a blog on Danish NLP[13], NLP introduction talks, and a demonstration page to show some of the models in action.[14]

6 A community for Danish NLP

The toolkit has so far benefited from bug reports, bug fixes, and suggestions for improvements from contributors through our GitHub repository.[15] The ambition is to have an even stronger community contributing to the toolkit with new models and datasets. The ambition is that the toolkit in time becomes more community-driven.

It is also within the project's scope to contribute to NLP frameworks to enable Danish's direct support. Before DaNLP, spaCy did not support Danish since an open-source NE dataset was lacking. However, with DaNE, (Hvingelby et al., 2020) this was fulfilled and is now part of spaCy.[16] The Flair tagging models for PoS and NER trained as part of DaNLP are now also available directly trough Flair.[17]

Nevertheless, the need for improving Danish NLP goes beyond a toolkit. It seems like the timing is opportune; currently, parties from academia and industry in Denmark are starting collaboration. Examples are recent, open-source models released by companies[18] and a large cross-

[13] https://medium.com/danlp
[14] https://danlp-demo.alexandra.dk
[15] https://github.com/alexandrainst/danlp
[16] https://spacy.io/models/da
[17] https://github.com/flairNLP/flair/blob/master/resources/docs/TUTORIAL_2_TAGGING.md
[18] https://github.com/sarnikowski/

collaboration on a large Danish text corpus named Gigaword by Strømberg-Derczynski et al. (2020). To strengthen the community around Danish NLP, the DaNLP project have gathered both companies in front of the field and researchers from Danish Universities (the Danish Technical University, the University of Copenhagen, and the IT University of Copenhagen) for network meetings to discuss and collaborate on Danish NLP [19]. One of the major identified challenges is how to gather and share data safely concerning privacy and GDPR.[20]

7 Concluding remarks

DaNLP is a new toolkit to make Danish NLP more applicable to industry. With this aim, the DaNLP project has been engaged in dialogues with industry, knowledge sharing and community building with academia and professionals. The hope is to continue working with a stronger community and inspire similar projects in other low to medium resource languages.

Acknowledgments

We want to thank the DaNLP team and Leon Derczynski, and our collaborators: the Danish Broadcasting Corporation and Infomedia. This work is supported by two performance contracts funded by the Danish Ministry of Higher Education and Science: "Dansk for Alle" and "Digital sikkerhed, etik og dataetik". Maria Barrett is supported by a research grant (34437) from VILLUM FONDEN.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, pages 1638–1649.

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual NLP. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192, Sofia, Bulgaria. Association for Computational Linguistics.

Apache Software Foundation. 2014. OpenNLP Natural Language Processing Library. Http://opennlp.apache.org/.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Antal van den Bosch, Bertjan Busser, Sander Canisius, and Walter Daelemans. 2007. An efficient memory-based morphosyntactic tagger and parser for Dutch. *LOT Occasional Series*, 7:191–206.

Gosse Bouma, Gertjan van Noord, and Robert Malouf. 2001. Alpino: Wide-coverage computational analysis of Dutch. In *Computational Linguistics in the Netherlands 2000*, pages 45–59. Brill Rodopi.

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

Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Leon Derczynski, Camilla Vilhelmsen Field, and Kenneth S Bøgh. 2014. DKIE: Open source information extraction for Danish. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 61–64.

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.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414.

danish_transformers/tree/main/electra
https://github.com/botxo/nordic_bert

[19] To join: https://danlp.alexandra.dk/

[20] https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016R0679&from=EN

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in python.

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. 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)*, page 157.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel S 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 5807–5812.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293.

Britt Keson. 1998. Vejledning til det danske morfosyntaktisk taggede parole-korpus. *Parole report, Det Danske Sprog- og Litteraturselskab (DSL)*.

Andreas Kirkedal, Barbara Plank, Leon Derczynski, and Natalie Schluter. 2019. The lacunae of Danish natural language processing. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 356–362.

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.

Hrafn Loftsson and Eiríkur Rögnvaldsson. 2007. IceNLP: A natural language processing toolkit for Icelandic. In *Eighth Annual Conference of the International Speech Communication Association*.

Edward Loper and Steven Bird. 2002. NLTK: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, pages 63–70.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26:3111–3119.

Emily Öhman, Kaisla Kajava, Jörg Tiedemann, and Timo Honkela. 2018. Creating a dataset for multilingual fine-grained emotion-detection using gamification-based annotation. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 24–30.

Vilhjálmur Þorsteinsson, Hulda Óladóttir, and Hrafn Loftsson. 2019. A wide-coverage context-free grammar for Icelandic and an accompanying parsing system. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1397–1404.

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. Association for Computational Linguistics.

Bolette Sandford Pedersen, Sanni Nimb, Jørg Asmussen, Nicolai Hartvig Sørensen, Lars Trap-Jensen, and Henrik Lorentzen. 2009. DanNet: the challenge of compiling a wordnet for Danish by reusing a monolingual dictionary. *Language resources and evaluation*, 43(3):269–299.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Uwe Quasthoff, Matthias Richter, and Christian Biemann. 2006. Corpus portal for search in monolingual corpora. In *LREC*, pages 1799–1802.

Radim Řehůřek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA.

Marta Sabou, Kalina Bontcheva, Leon Derczynski, and Arno Scharl. 2014. Corpus annotation through crowdsourcing: Towards best practice guidelines. In *LREC 2014*, pages 859–866.

Nina Schneidermann, Rasmus Hvingelby, and Bolette Sandford Pedersen. 2020. Towards a gold standard for evaluating Danish word embeddings. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4754–4763.

Dansk Sprognævn. 2019. Dansk sprogteknologi i verdensklasse.

Milan Straka, Jan Hajic, and Jana Straková. 2016. Udpipe: trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4290–4297.

Leon Strømberg-Derczynski, Rebekah Baglini, Morten H Christiansen, Manuel R Ciosici, Jacob Aarup Dalsgaard, Riccardo Fusaroli, Peter Juel Henrichsen, Rasmus Hvingelby, Andreas Kirkedal, Alex Speed Kjeldsen, et al. 2020. The Danish gigaword project. *arXiv preprint arXiv:2005.03521*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45. Association for Computational Linguistics.