Introducing the NLU Showroom: A NLU Demonstrator for the German Language

Dennis Wegener
Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany

Sven Giesselbach
Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany

Niclas Doll
Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany

Heike Horstmann
Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany

Abstract

We present the NLU Showroom, a platform for interactively demonstrating the functionality of natural language understanding models with easy to use visual interfaces. The NLU Showroom focuses primarily on the German language, as not many German NLU resources exist. However, it also serves corresponding English models to reach a broader audience. With the NLU Showroom we demonstrate and compare the capabilities and limitations of a variety of NLP/NLU models. The four initial demonstrators include a) a comparison on how different word representations capture semantic similarity b) a comparison on how different sentence representations interpret sentence similarity c) a showcase on analyzing reviews with NLU d) a showcase on finding links between entities. The NLU Showroom is build on state-of-the-art architectures for model serving and data processing. It targets a broad audience, from newbies to researchers but puts a focus on putting the presented models in the context of industrial applications.

2012 ACM Subject Classification Applied computing → Document management and text processing

Keywords and phrases Natural Language Understanding, Natural Language Processing, NLU, NLP, Showroom, Demonstrator, Demos, Text Similarity, Opinion Mining, Relation Extraction

Digital Object Identifier 10.4230/OASIcs.LDK.2021.28

Funding This research has been funded by the Federal Ministry of Education and Research of Germany as part of the competence center for machine learning ML2R (01IS18038B).

1 Introduction

Natural language processing (NLP) and understanding (NLU) have gained a lot of interest over the past years. In the following, we will refer to both NLP and NLU as NLU. Recent developments in the field have led to significant improvements for many use cases, enabling the usage of NLU Models in real world application. This applies especially for English, for which a lot of large annotated corpora are available. However, this leaves a gap for other languages such as German.

The goal of the NLU Showroom is to demonstrate the capabilities of NLU models for the German language in order to raise interest from people outside of the scientific domain for integrating NLU models in their applications. The demonstrators provided with the NLU Showroom should serve as showcases for the capabilities of state-of-the-art NLU models and their limitations for a variety of tasks. In the current version we provide demonstrators for a) comparing how different word representation models capture semantic similarity of words when trained on the same corpus, i.e. the German Wikipedia b) Comparing how different sentence representation models interpret sentence similarity when trained on the same corpus,

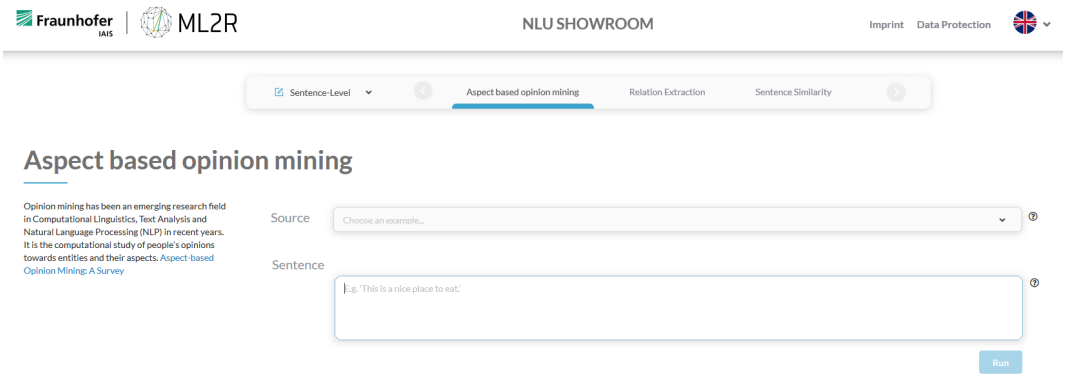

Figure 1 The interface of the NLU Showroom lets the user navigate through different demos and lets them interactively explore NLU model capabilities and differences. The users can enter their own prompt or select example inputs.

also Wikipedia c) showcasing how multiple state-of-the-art BERT [9] models can be used to analyze reviews and extract and link aspects and opinions d) showcase how state-of-the-art relation extraction models, extract relations for preselected entities.

The heart of the NLU Showroom is a web-based frontend, which allows the user to interactively explore NLU models in multiple demonstrators (see Figure 1). At it's core, the NLU Showroom has been built on a strong architectural base, so that it can easily be extended with additional models based on PyTorch and TensorFlow as well as other non-neural models. In the following we will refer to NLU tasks such as named entity recognition as "tasks", the models used to solve the tasks as "models" and the interactive demo of a model regarding a task as "demonstrator".

2 Related platforms and demos

A handful of NLU demo platforms already exist. Most of them are demonstrating natural language analysis packages and appeal to experts in the field of NLU rather than to people who want to understand how to utilize NLU in their applications.

A very recent addition is Stanfords Stanza [21]. It is a Python natural language analysis package which is also presented in an online demo [5]. The demo includes the models provided in the package, namely for Part-of-Speech Tagging, Lemmatization, Named Entity Recognition (NER) and Dependency Parsing models for a variety of languages (including German). All of these models represent important steps in the analysis and understanding of texts. However, with the exception of NER, these models address rather low-level tasks and clearly tackle NLU practitioners. They might be too in-depth for people outside of the NLU domain. The same applies for the four demos of the python natural language toolkit (NLTK) [4]. The demos include word tokenizers, stemming in 17 languages, part of speech tagging with 22 different part of speech taggers and finally sentiment analysis, which uses text classification to determine sentiment polarity in 3 languages.

The IBM Watson NLU demo on Text Analysis [3] includes the extraction of entities, keywords, concepts and relations, the classification of sentiment, emotions and categories, and linguistic analysis of semantic roles and syntax. The demo presents prepared examples from different industrial domains and allows to enter own text or URL pointers. However, in contrast to the NLU Showroom, it provides no information about the models used for the different tasks.

The European Language Grid [22] is a platform for supporting language technologies in the European market. It aims at delivering not only datasets but also services and demos for EU official languages and EU languages without official status. The services include a variety of NLP tasks, also for the German language. However, the services seem to be rather focused on NLU practitioners and scientists than on the general public.

The same applies to the Hugging Face Transformer framework [27]. Hugging Face is an open source framework providing pretrained models for a variety of NLP tasks based on Transformer models only. It aims at data scientists and researchers and is primarily focused on the sharing of models and data.

The demonstrator the NLU Showroom shares most similarities with is the one from AllenNLP [10]. AllenNLP is an open-source natural language processing platform for building state of the art models, which also includes a demo on the functionality of NLU models [1]. It includes demonstrations for many NLU tasks, including those that the NLU Showroom aims at. However, the demos and models are only available in the English language.

3 Tasks and Models

For the initial release of the NLU Showroom we selected a variety of tasks which demonstrate how NLU models work, what they are capable of, what their limitations are and how they can be used. We designed demonstrators which easily show how different models understand natural language. The tasks in the demos are of different granularity. Some of them represent word-level tasks, while others are on a sentence basis. In addition, the tasks are of different complexity levels, e.g. single models vs several models that are combined to complete a certain task. The showroom contains short explanations of all the tasks, the models and the data they were trained on.

In the following, we will present the tasks and describe the underlying models. All models were trained on German and English data sets.

3.1 Word-Level

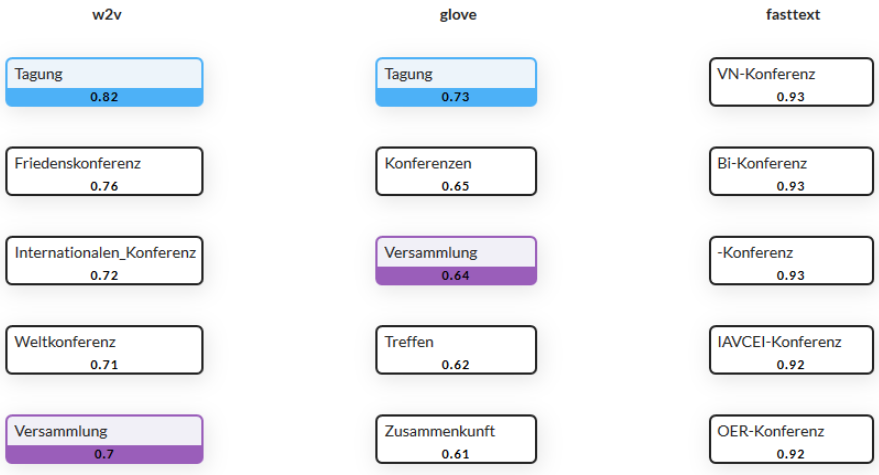

Figure 2 An example output of the word similarity task for the word “Konferenz”. The user can see how 3 models evaluate the similarity in a different way. Words that are highlighted in colours are returned by multiple models. The numbers represent the cosine similarity to the query word.

The NLU Showroom currently contains one word-level demonstrator. The demonstrator on the word-level basis shows how different word representation models capture semantic similarity when trained on the same corpus with similar hyper-parameters. We trained multiple distributed embedding models on the same pre-processed versions of the German and English Wikipedia. The models we trained are Word2Vec [15], Glove [18] and FastText [8]. We have two intentions with this demonstrator: 1) Show that these representations capture meaningful similarities and can be used e.g. to enrich search functions or ontologies 2) Show that these models capture similarities in different ways – e.g. that the influence of using character n-grams in FastText heavily influences what is considered similar when compared to Word2Vec. Reason 2) also motivates why we did not use pretrained embeddings since the data they were pre-processed with was most likely from multiple sources. For the same reason all models have been trained with roughly the same parameters. In detail, all embeddings have been trained with embedding dimensionality set to 300, a context window size of 5 and 5 negative samples. Our word similarity demo (see Figure 2) lets users input query words and phrases and explore the nearest neighbors that the models obtain. It highlights similarities and differences between the responses of the different models. An interesting observation is that despite their theoretical similarity [14] Word2Vec and Glove produce surprisingly different results. It is also easily observable that FastText focuses more on morphological similarities. In the future we will add further word representation models and let the users select which models to compare.

3.2 Sentence-Level

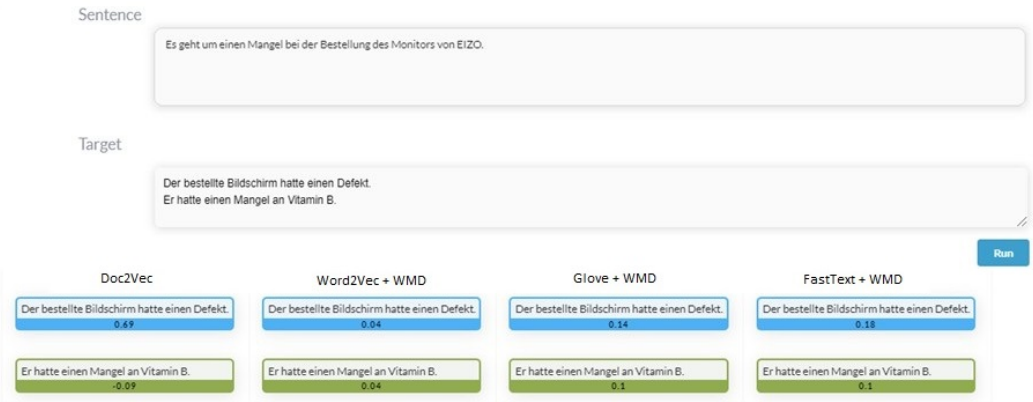

Figure 3 An exemplary output of the sentence similarity task. The user can enter a source sentence and several target sentences. As result we show the similarities of the target sentences to the source sentence computed by the different models. A higher number means greater similarity. Across models the same sentence is highlighted in the same color for easier comparison.

In our initial release of the NLU Showroom we integrate demonstrators for 3 different sentence-level tasks. The first demo displays a sentence similarity task, similar to the word-similarity task. Users can input a source sentence and target sentences and then our models compare the similarity of the source sentence to the different target sentences. Many text based processes require the need to compare texts and identify similar texts or passages. We intend to highlight how the choice of model can influence the similarities and that the models rely on more than simple string or keyword matching. Our demonstrator currently includes 4 different models. We trained a Doc2Vec [13] model on the German and English Wikipedia

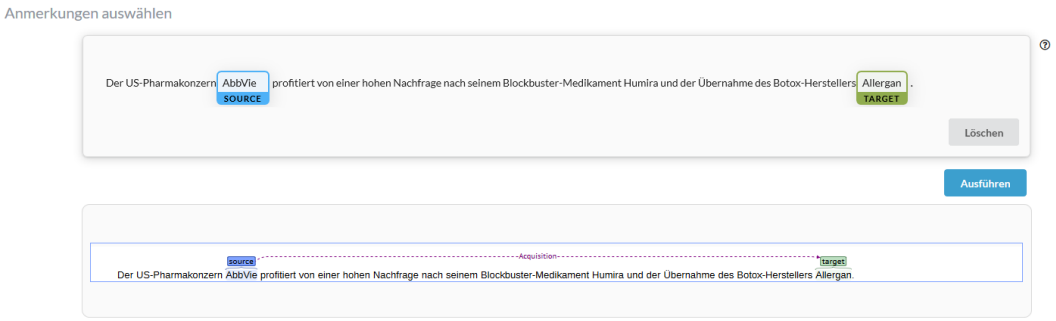

Figure 4 The output of our relation extraction and classification demonstrator. The demo requires 2 steps: 1) The user inputs a sentence 2) the user highlights the entities which should be checked for a possible relation. If a relation is detected, the name of the relation is returned.

and use the Word Mover's Distance [12] together with the 3 different word representation models mentioned in the word-level task. The demo displays the similarity rankings of the different target sentences to the source sentence. Figure 3 shows an example call of the sentence similarity task. Notably all models understand that target sentence 1 is closer to the source sentence, even though target sentence 2 and the source sentence share more words. We will integrate new Transformer-based similarity measures soon.

Our second sentence-level demo, demonstrates a relation extraction and classification model. Here we use a BERT-based model [24] that was trained on the German Smartdata Corpus [23] and the English SemEval2010 Task8 data set [11]. In the demo, the user can input a sentence and specify entities which should be checked for relations (see Figure 4). The model then visualizes the relation class, if a relation has been detected.

Third, we present an aspect-based opinion mining demo. This demo actually combines three models, trained on German and English Yelp reviews. It starts by extracting aspects and opinions from the text using a BERT-based architecture [25]. Another BERT-based model is applied to compute sentiment labels. Lastly we link aspects and opinions by finding shortest paths in the dependency tree. The dependency tree is build using StanfordNLP's dependency parser [20].

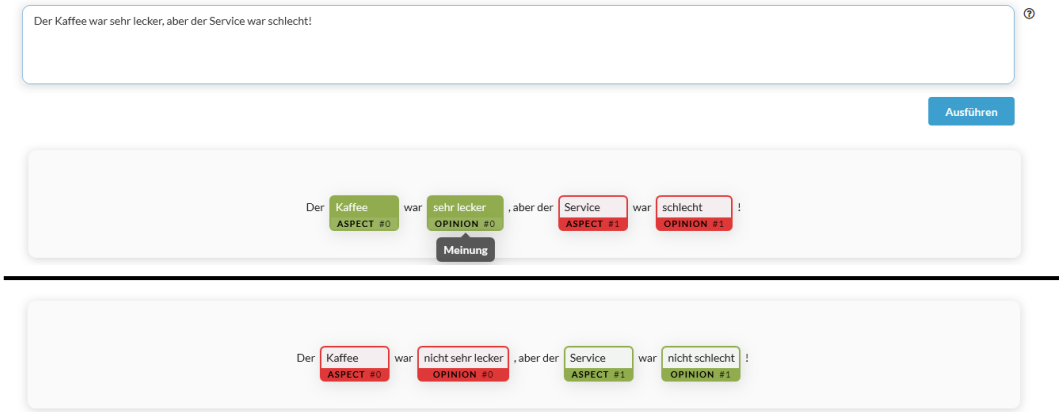

Figure 5 Exemplary outputs of the opinion mining task. The first sentence includes two different sentiments (aspect-opinion pairs). Aspects and opinions are linked via a number next to their description and links are highlighted when hovered over. Green represents positive and red negative sentiment. In the second sentence we negated the opinions.

The English models were trained on the SemEval2014-Task4 data set [19], which originally only consisted of aspect labels and their sentiments, but was extended to also annotate opinion labels [26]. For the German language we crawled and manually annotated reviews from Yelp to create a dataset, which we aim to publish in the future. The demo asks the users to enter a restaurant review and will then output the extracted aspects and opinions as well as the links between them and the according sentiments (see Figure 5). It serves as a demonstration of what is possible in automated review analysis with state-of-the-art models. Experiments with the models show that it is rather resistant to spelling mistakes, can handle negations and multiple aspect and opinion pairs. However, it fails when multiple opinions are linked to the same aspect.

4 Technical Architecture

The NLU Showroom demonstrates state-of-the art NLU models. These models are typically created with the help of modern open source frameworks, such as TensorFlow [7] and PyTorch [17]. These frameworks also already include components for serving the models. As part of TensorFlow Extended (TFX), TensorFlow Serving [16] is a serving system for machine learning models which is designed for production environments. It provides integration with TensorFlow models and can be easily extended to serve other types of models and data. TorchServe [6] is a light-weight tool for deploying and serving PyTorch models. Since the NLU Showroom is focused on quickly demonstrating latest models from ML research, we integrated these two open source frameworks in the backend of our NLU Showroom. The training process of the models is not explicitly addressed by our architecture and is performed offline in a GPU computing cluster.

In detail, the NLU Showroom consists of a frontend and several backend components. The frontend is a web application that is based on frontend frameworks such as Node.js and React. It includes the actual web pages, the controls to interact with the models and the different visualizations for the model results. The frontend communicates with the backend part via REST services. The backend includes several components which are mainly responsible for serving the models. As we already stated, we use TensorFlow Serving [16] and TorchServe [6] for serving models from the different frameworks. In addition, a webserver that serves as wrapper for the model serving components is responsible for transforming the model results into visualization content and also for wrapping several models for a single task, e.g., for the task on aspect based opinion mining where 3 models are combined.

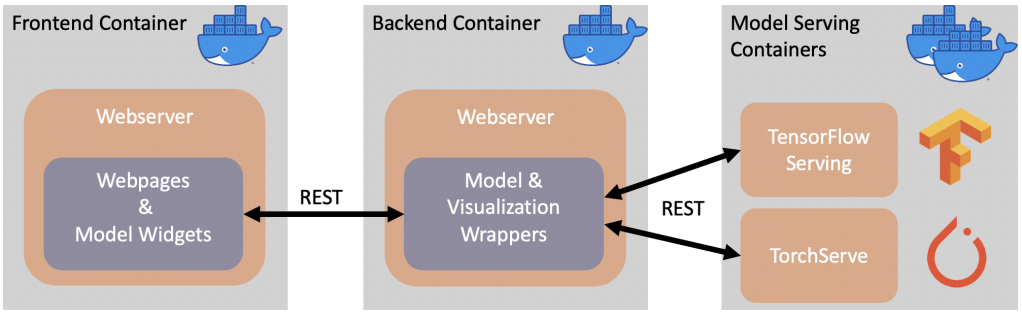

Figure 6 The technical architecture of the NLU Showroom.

The whole architecture is based on Docker [2] in order to easily isolate the different components. This allows us to take standard open source components as TensorFlow Serving and TorchServe without any modifications, and isolate our wrapper and glue code in a single

separate component (the web server). The containerization via Docker also allow us to later deploy the system into an auto-scaling production environment. Figure 6 gives an overview over the technical architecture.

New models can be integrated in an easy way. They just have to be deployed into the TorchServe or the TensorFlow Serving container. If the new model refers to an existing task, there are just minor adaptions needed on the wrapper backend component to forward the model output to the frontend. If it refers to a new task, there is the need to develop a suitable widget for it. In addition, the wrapper needs to be extended for tasks that include several models or build on models that are not deployed into TorchServe or TensorFlow Serving but need custom integration. Currently, the set of tasks and models is curated by the project team. In the future we might open the NLU Showroom to further contributors.

5 Conclusion

We present the NLU Showroom, a platform for demonstrating NLU models. The platform aims at people outside of the NLU community, who are intending to use NLU in their applications and products. In the current version, the NLU Showroom includes state-of-the art models for word and sentence similarity tasks, aspect and opinion mining and relation extraction, with more demos and models to follow. The platform builds on state-of-the-art open source components such as TensorFlow Serving, PyTorch and Docker.

In addition to releasing more demos we aim to link the showroom in the blog of the Competence Center for Machine Learning Rhine-Ruhr (ML2R). In the blog we will give detailed explanations about the models, tasks and data sets used to create the demos of the NLU Showroom and describe how these models can and have been used in projects and products.

References

1 AllenNLP demo. `https://demo.allennlp.org`. Accessed: 2021-02-02.

2 Docker. `https://www.docker.com/`. Accessed: 2021-02-02.

3 IBM Watson natural language understanding - text analysis. `https://www.ibm.com/demos/live/natural-language-understanding/self-service/home`. Accessed: 2021-02-02.

4 NLTK text processing demo. `http://text-processing.com/demo/`. Accessed: 2021-02-02.

5 Stanza online demo. `http://stanza.run/`. Accessed: 2021-02-02.

6 TorchServe. `https://github.com/pytorch/serve`. Accessed: 2021-02-02.

7 Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. URL: `https://www.tensorflow.org/`.

8 Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *CoRR*, abs/1607.04606, 2016. `arXiv:1607.04606`.

9 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, 2019. Association for Computational Linguistics. `doi:10.18653/v1/N19-1423`.

10 Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. AllenNLP: A deep semantic natural language processing platform. *arXiv*, 2017. arXiv:1803.07640.

11 Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, 2010. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/S10-1006.

12 Matt Kusner, Y. Sun, N.I. Kolkin, and Kilian Weinberger. From word embeddings to document distances. *Proceedings of the 32nd International Conference on Machine Learning (ICML 2015)*, pages 957–966, January 2015.

13 Quoc V. Le and Tomas Mikolov. Distributed representations of sentences and documents. *CoRR*, abs/1405.4053, 2014. arXiv:1405.4053.

14 Omer Levy and Yoav Goldberg. Neural word embedding as implicit matrix factorization. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 2177–2185, Cambridge, MA, USA, 2014. MIT Press.

15 Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546, 2013. arXiv:1310.4546.

16 Christopher Olston, Noah Fiedel, Kiril Gorovoy, Jeremiah Harmsen, Li Lao, Fangwei Li, Vinu Rajashekhar, Sukriti Ramesh, and Jordan Soyke. Tensorflow-serving: Flexible, high-performance ML serving. *CoRR*, abs/1712.06139, 2017. arXiv:1712.06139.

17 Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

18 Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. URL: http://www.aclweb.org/anthology/D14-1162.

19 Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, 2014. Association for Computational Linguistics. doi:10.3115/v1/S14-2004.

20 Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning. Universal dependency parsing from scratch. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 160–170, Brussels, Belgium, October 2018. Association for Computational Linguistics. URL: https://nlp.stanford.edu/pubs/qi2018universal.pdf.

21 Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 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*, 2020. URL: https://nlp.stanford.edu/pubs/qi2020stanza.pdf.

22 Georg Rehm, Maria Berger, Ela Elsholz, Stefanie Hegele, Florian Kintzel, Katrin Marheinecke, Stelios Piperidis, Miltos Deligiannis, Dimitris Galanis, Katerina Gkirtzou, Penny Labropoulou, Kalina Bontcheva, David Jones, Ian Roberts, Jan Hajič, Jana Hamrlová, Lukáš Kačena, Khalid Choukri, Victoria Arranz, Andrejs Vasiļjevs, Orians Anvari, Andis Lagzdiņš, Jūlija Meļņika, Gerhard Backfried, Erinç Dikici, Miroslav Janosik, Katja Prinz, Christoph Prinz,

Severin Stampler, Dorothea Thomas-Aniola, José Manuel Gómez-Pérez, Andres Garcia Silva, Christian Berrío, Ulrich Germann, Steve Renals, and Ondrej Klejch. European language grid: An overview. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3366–3380, Marseille, France, 2020. European Language Resources Association. URL: https://www.aclweb.org/anthology/2020.lrec-1.413.

23 Martin Schiersch, Veselina Mironova, Maximilian Schmitt, Philippe Thomas, Aleksandra Gabryszak, and Leonhard Hennig. A german corpus for fine-grained named entity recognition and relation extraction of traffic and industry events. In *Proceedings of the 11th International Conference on Language Resources and Evaluation. International Conference on Language Resources and Evaluation (LREC-18), 11th, May 7-12, Miyazaki, Japan*. European Language Resources Association, 2018.

24 Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the blanks: Distributional similarity for relation learning. *CoRR*, abs/1906.03158, 2019. arXiv:1906.03158.

25 Chi Sun, Luyao Huang, and Xipeng Qiu. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. *CoRR*, abs/1903.09588, 2019. arXiv:1903.09588.

26 Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 616–626, Austin, Texas, 2016. Association for Computational Linguistics. doi:10.18653/v1/D16-1059.

27 Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771, 2019. arXiv:1910.03771.