From open parallel corpora to public translation tools : The success story of OPUS

Tiedemann, Jörg

University of Göteborg
2022-11

Tiedemann , J 2022 , From open parallel corpora to public translation tools : The success story of OPUS . in E Volodina , D Dannélls , A Berdicevskis , M Forsberg & S Virk (eds) , LIVE and LEARN : Festschrift in honor of Lars Borin . Research Reports from the Department of Swedish, Multilingualism, Language Technology , no. GU-ISS-2022-03 , University of Göteborg , Göteborg , pp. 133-138 .

http://hdl.handle.net/10138/351496

From open parallel corpora to public translation tools: The success story of OPUS

Jörg Tiedemann
Department of Digital Humanities Language Technology
University of Helsinki, Finland
jorg.tiedemann@helsinki.fi

Abstract

This paper describes the success of OPUS, starting from a small side-project but leading to a full-fledged ecosystem for training and deploying open machine translation systems. We briefly present the current state of the framework focusing on the mission of increasing language coverage and translation quality in public translation models and tools that can easily be integrated in end-user applications and professional workflows. OPUS now provides the biggest hub of freely available parallel data and thousands of open translation models have been released supporting hundreds of languages in various combinations.

1 Introduction

The starting point of OPUS is clearly connected to Uppsala and the language technology research group at the former department of linguistics. Work with parallel corpora has been pushed by projects on machine translation (Tjong Kim Sang, 1999) and multilingual corpus-driven linguistics and lexicography (Borin, 1998; Borin, 2002). The significant value of aligned multilingual data sets had been recognized by the leading researchers in the group and various resources came out of their efforts together with applications in translation studies, bilingual lexicon induction and machine translation development (Borin, 2000a; Borin, 2000b; Sågvall Hein et al., 2002). Inspired by those projects, OPUS filled the gap of public data sets that can be freely shared and used in research and development. Initially starting with software localization data, OPUS slowly grew into a massive collection of parallel translation data covering hundreds of languages and thousands of language pairs coming from a wide variety of domains.

The mission of OPUS was clear from the beginning: Data sets in the collection shall be open and free and support reproducible science to push cross-lingual NLP research and machine translation in particular. The essential principle is to provide a consistent interface to data sets that are readily prepared for further work without losing information from the original source. Wide language coverage has been a goal from the start with a complete alignment across all languages included.

The collection now represents a crucial foundation for wide-coverage machine translation. Taking advantage of the huge resource, we launched OPUS-MT (Tiedemann & Thottingal, 2020), an initiative to systematically exploit the data set to train open neural machine translation (MT) models that can be shared and re-used as well. The project tackles the growing responsibility of language technology providing essential tools for fair information access without language barriers and avoiding commercial exploitation. Our focus is on transparency and the paper describes our efforts in building the infrastructure that enables the use of free and independent machine translation in end-user applications and professional workflows.

Below, we briefly provide the background on OPUS and present tools for finding and processing the data. We then introduce OPUS-MT and its components before discussing the integration of pre-trained translation models in development platforms, end-user applications and translation workflows. Finally,

Jörg Tiedemann. 2022. From open parallel corpora to public translation tools: The success story of OPUS. In Volodina, Dannélls, Berdicevskis, Forsberg and Virk (editors), *Live and Learn – Festschrift in honor of Lars Borin*, pages 133–138. Available under CC BY 4.0

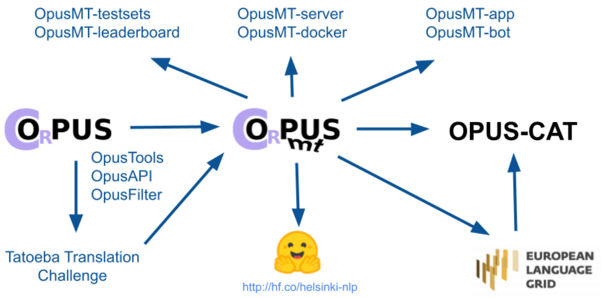


Figure 1: OPUS and OPUS-MT and its connections to other components, platforms and applications.

we also present the importance of benchmarking and monitoring the progress and briefly mention ongoing work on scaling up language coverage and optimizing translation models in terms of speed and applicability. Figure 1 illustrates the connections between various components.

2 OPUS – The Open Parallel Corpus

OPUS[1] has been a major hub for parallel corpora since 2004 (Tiedemann & Nygaard, 2004; Tiedemann, 2009; Tiedemann, 2012). The current release covers over 600 languages compiled into sentence-aligned bitexts for more than 40,000 language pairs. Over 20 billion sentences and sentence fragments correspond to 290 billion tokens and the data set contains about 12 TB of compressed files. Despite the typical Zipfian distribution, there are over 300 language pairs with more than one million sentence pairs, a good base for high quality machine translation.

OPUS tries to follow a consistent format with a simple standalone XML format for language content and standoff annotation in XCES Align to annotate links between translated sentences. The latter enables a space-efficient way of storing bilingually-aligned multilingual data sets without duplicating essential content. For convenience, other common data formats are generated from the native OPUS format including plain text versions with aligned sentences on corresponding lines and translation memory exchange files (TMX) that are common in professional translation platforms. Additionally, OPUS also releases token frequency counts, word alignment files and rough bilingual dictionaries extracted from automatically aligned bitexts.

Recently, we also released a compilation of the data under the label of the *Tatoeba Translation Challenge*[2], TTC for short (Tiedemann, 2020). The purpose of this release is to provide a streamlined collection for MT training pipelines. The latest release of the TTC includes 29 billion translation units in 3,708 bitexts covering 557 languages altogether. We made an effort to unify different sources, to improve the consistency in language labeling and to remove noise and duplicates. Dedicated development and test sets are also provided to make the application of TTC as straightforward as possible in standard machine learning setups.

2.1 Finding and processing OPUS data with the OPUS-API and OpusTools

An important ingredient for OPUS is automation. Making resources available requires efficient ways of finding and accessing them. The OPUS-API[3] provides an online API for searching resources and enables

[1] `https://opus.nlpl.eu`
[2] `https://github.com/Helsinki-NLP/Tatoeba-Challenge`
[3] `https://opus.nlpl.eu/opusapi/`

queries for specific languages and corpora. It provides the essential information about released data sets and returns download links to fetch data from the external data storage. The API responds in simple JSON format, which can easily be used programmatically when looking for resources.

We make use of the OPUS-API ourselves with the implementation of the OpusTools package[4] (Aulamo et al., 2020a). This software library provides a Python interface with methods for locating, downloading and converting OPUS data sets. Command-line scripts such as *opus_read* provide convenient functions to query the database and to fetch data from the original storage. Furthermore, the tools read from compressed release-packages and can be used to convert data sets into various formats such as TMX and plain text on the fly. Sentence alignments can also be filtered based on alignment confidence score, alignment type or language flag. For the latter, the package includes tools for automatic language identification.

2.2 Cleaning parallel data with OpusFilter

OpusFilter[5] (Aulamo et al., 2020b) integrates the functionality provided by OpusTools and the OPUS-API but adds a modular system for filtering and preparing parallel data sets. It provides a wide variety of modules for data preparation and noise reduction. A YAML configuration file defines the pipeline to transform raw corpus files to clean training and test set files. The same pipeline can be generalized over multiple language pairs. The toolbox can easily be extended and currently supports different kinds of segment-level processing steps such as tokenization and subword splitting as well as filters based on automatic language identification, word alignment scores, language models and sentence embeddings. Furthermore, scores can be analyzed and visualized, and custom classifiers can be trained to make domain-specific filter decisions (Vázquez et al., 2019).

3 Machine translation with OPUS-MT

The natural next step after collecting and compiling parallel data is to systematically exploit them in learning machine translation models. OPUS-MT[6] aims to provide training pipelines and solutions for deploying MT models derived from OPUS data. The goal is to develop a major hub for open state-of-the-art models with a large language coverage and straightforward use in end-user applications and further research and development. The framework is based on Marian, an efficient implementation of neural machine translation (NMT) in pure C++ and with minimal dependencies (Junczys-Dowmunt et al., 2018).

OPUS-MT training pipelines come in the form of makefile recipes that enable massive and systematic experiments on high-performance computing facilities. Automation provided by the recipes cover all necessary sub-tasks for preparing data sets, training models, testing their performance and finally releasing pre-trained NMT models. Special care has been taken to allow the creation of multilingual models that support more than one language as input or output. The recipes transparently handle different language combinations and combine data sets as necessary adding language flags if required (Johnson et al., 2017). Subword segmentation using SentencePiece (Kudo & Richardson, 2018) is fully integrated, and automatic word alignment (Östling & Tiedemann, 2016) can be used to train transformer models with guided alignment features. Batch jobs can easily be created to run on SLURM-based task management systems.

OPUS-MT further provides pipelines for data augmentation using back-translation (Sennrich et al., 2016) or pivot-based triangulation. Fine tuning is also supported in order to adapt to specific domains, user-specific data sets or selected language pairs in mulitlingual models.

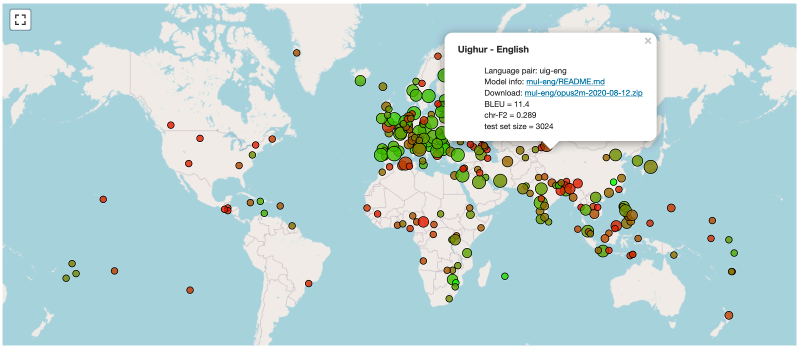
3.1 Integrating OPUS-MT

Important for the success of pre-trained models is the ease of use and deployment. OPUS-MT strives to make the models accessible and useful for a wide range of users. Substantial efforts have been made to provide simple deployment procedures and integration routines for all our models.

[4] `https://github.com/Helsinki-NLP/OpusTools`

[5] `https://github.com/Helsinki-NLP/OpusFilter`

[6] `https://github.com/Helsinki-NLP/OPUS-MT`



Uighur - English

Language pair: uig-eng
Model info: mul-eng/README.md
Download: mul-eng/opus2m-2020-08-12.zip
BLEU = 11.4
chr-F2 = 0.289
test set size = 3024

Figure 2: Language coverage of translation models visualized on an interactive map. Geolocations of languages are taken from Glottolog and dot colors indicate the translation quality in terms of an automatic evaluation metric measured on the Tatoeba test set in this case on a scale from green (best) to red (worst). Smaller circles refer to smaller, less reliable test sets.

First of all, we provide methods to create translation servers using web applications that provide service APIs through web sockets and requests. Servers can easily be configured using JSON files and the API also uses JSON for communication. Multiple translation servers can be combined and accessed via the same interface and caching is implemented to decrease the workload of the server. All pre-trained models we release can be integrated in the server solutions we offer.

Another important integration is the conversion of the native Marian NMT models to PyTorch, which opens up their use in a wide range of applications through the popular transformers library provided by Huggingface.[7] Conversion scripts are available to prepare OPUS-MT models for the public model hub making them available to the NLP community and also accessible through the online inference API.

Similarly, we also integrate OPUS-MT models in the European Language Grid (ELG). Dockerized OPUS-MT servers run on the ELG platform making it possible to directly access translation models from the ELG cloud services and the APIs provided by the infrastructure. The same docker images can also be downloaded from DockerHub and may run locally or on other cloud infrastructures.

Addressing the needs of professional translators is done by OPUS-CAT,[8] a collection of tools and plugins that add the power of OPUS-MT to computer-assisted translation (CAT) workflows in Trados Studio, memoQ, and OmegaT. In some CAT tools, such as Wordfast, OPUS-CAT can be used by connecting directly to its API through a custom MT provider functionality. OPUS-CAT also includes a Chrome browser extension, which makes it possible to use OPUS-MT in browser-based CAT tools. The Chrome extension currently supports Memsource[9] and XTM.[10] Different to other solutions, OPUS-CAT runs MT locally and does not require to send data to any external service. This has huge advantages in terms of data privacy and security and also enables fine-tuning of local translation engines on custom data without compromizing data safety.

3.2 Benchmarks and evaluation

Monitoring language coverage and translation quality is important to keep track of the progress in our mission to improve language support and cross-lingual information accessibility using open MT solutions. Therefore, we systematically run benchmarks on all our models using a wide range of test sets coming from established evaluation campaigns. Automatic evaluation is certainly not sufficient but still

[7] `https://huggingface.co/transformers/`

[8] `https://helsinki-nlp.github.io/OPUS-CAT/`

[9] `https://www.memsource.com/`

[10] `https://xtm.cloud/`

provides a good indication of quality especially if several benchmarks are used in parallel instead of relying on single test sets.

We aim at a comprehensive collection of benchmarks[11] and results are stored in a public repository,[12] which can be explored in a public leaderboard.[13] Translation results are also kept in the same repository to make it possible to run further qualitative studies on actual output of each model. Finally, we also create dynamic maps that visualize language coverage according to geographic locations of languages supported by OPUS-MT (see Figure 2).

4 Conclusions

Above, we have shown how OPUS developed from a small data collection initiative to a mature ecosystem for research on large coverage machine translation. All the components connected to OPUS provide a complete framework for systematic experiments and state-of-the-art neural MT development. The main building blocks refer to data collection, data curation, model training, system evaluation as well as deployment and MT integration tasks. The data collection itself is extensive but the coverage of released MT models is also impressive already. A lot of further work is on-going including the implementation of modular multilingual machine translation and the development of speed-optimized compact translation models using various kinds of knowledge distillation and quantization. Further integration into end-user applications on various devices are planned as well and translation quality and language coverage are constantly improved.

References

Mikko Aulamo, Umut Sulubacak, Sami Virpioja, & Jörg Tiedemann. 2020a. OpusTools and Parallel Corpus Diagnostics. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3782–3789, Marseille. European Language Resources Association.

Mikko Aulamo, Sami Virpioja, & Jörg Tiedemann. 2020b. OpusFilter: A Configurable Parallel Corpus Filtering Toolbox. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 150–156. Association for Computational Linguistics.

Lars Borin. 1998. ETAP: Etablering och annotering av parallellkorpus för igenkänning av översättningsekvivalenter (ETAP: Creating and annotating a parallel corpus for the recognition of translation equivalents). *ASLA Information*, 24(1):33–40.

Lars Borin. 2000a. ETAP project status report December 2000. Technical report, Uppsala University, Department of Linguistics.

Lars Borin. 2000b. You'll Take the High Road and I'll Take the Low Road: Using a Third Language to Improve Bilingual Word Alignment. In *COLING 2000 Volume 1: The 18th International Conference on Computational Linguistics*.

Lars Borin. 2002. Alignment and tagging. In *Parallel corpora, parallel worlds. Selected papers from a symposium on parallel and comparable corpora at Uppsala University, Sweden, 22-23 April, 1999*, Language and computers: studies in practical linguistics, pages 207–218. Amsterdam: Rodopi.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, & Jeffrey Dean. 2017. Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, & Alexandra Birch. 2018. Marian: Fast Neural Machine Translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne. Association for Computational Linguistics.

[11] `https://github.com/Helsinki-NLP/OPUS-MT-testsets`

[12] `https://github.com/Helsinki-NLP/OPUS-MT-leaderboard/`

[13] `https://opus.nlpl.eu/leaderboard/`

Taku Kudo & John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels. Association for Computational Linguistics.

Robert Östling & Jörg Tiedemann. 2016. Efficient word alignment with Markov Chain Monte Carlo. *Prague Bulletin of Mathematical Linguistics*, 106:125–146.

Anna Sågvall Hein, Eva Forsbom, Jörg Tiedemann, Per Weijnitz, Ingrid Almqvist, Leif-Jöran Olsson, & Sten Thaning. 2002. Scaling Up an MT Prototype for Industrial Use - Databases and Data Flow. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation, (LREC'2002)*, volume V, pages 1759–1766, Las Palmas de Gran Canaria.

Rico Sennrich, Barry Haddow, & Alexandra Birch. 2016. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin. Association for Computational Linguistics.

Jörg Tiedemann & Lars Nygaard. 2004. The OPUS Corpus - Parallel and Free: `http://logos.uio.no/opus`. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon. European Language Resources Association (ELRA).

Jörg Tiedemann & Santhosh Thottingal. 2020. OPUS-MT - Building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation (EAMT)*. European Association for Machine Translation.

Jörg Tiedemann. 2009. News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In N. Nicolov, K. Bontcheva, G. Angelova, & R. Mitkov, editors, *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia, Borovets.

Jörg Tiedemann. 2012. Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul. European Language Resources Association (ELRA).

Jörg Tiedemann. 2020. The Tatoeba Translation Challenge – Realistic Data Sets for Low Resource and Multilingual MT. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.

Erik Tjong Kim Sang. 1999. Aligning the Scania Corpus. *Working Papers in Computational Linguistics & Language Engineering*, 18.

Raúl Vázquez, Umut Sulubacak, & Jörg Tiedemann. 2019. The University of Helsinki Submission to the WMT19 Parallel Corpus Filtering Task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 294–300, Florence. Association for Computational Linguistics.