Global Under-Resourced MEdia Translation (GoURMET)

H2020 Research and Innovation Action
Number: 825299
D1.2 – Initial progress report on data gathering and augmentation

Nature	Report	**Work Package**	WP1
Due Date	30/06/2020	**Submission Date**	30/06/2020
Main authors	Felipe Sánchez-Martínez (UA)		
Co-authors	Miquel Esplà-Gomis (UA), Víctor M. Sánchez-Cartagena (UA), Barry Haddow (UEDIN), Radina Dobreva (UEDIN)		
Reviewers	Alexandra Birch (UEDIN)		
Keywords	data gathering, data augmentation, corpora, machine translation		
Version Control			
v0.8	**Status**	1st Draft	08/06/2020
v0.9	**Status**	2nd Draft	19/06/2020
v1.0	**Status**	Final	25/06/2020
v1.1	**Status**	Minor update	03/07/2020

Contents

Abstract

This deliverable reports the work conducted within workpackage WP1 on data gathering and data augmentation. It focus on three main tasks: crawling of monolingual and bilingual corpora from the web, processing data dumps provided by the user partners to obtain in-domain corpora mainly for testing, and the application and development of data augmentation techniques for generating synthetic training corpora.

1 Introduction

Workpackage WP1 focuses on the identification and collection on linguistic resources —morphological dictionaries, lexical disambiguation models, bilingual dictionaries, translation rules, etc.— for the languages of interest to GoURMET (task T1.1), the identification, collection and evaluation of monolingual and bilingual corpora (task T1.2) and the generation of synthetic data and lexical augmentation (task T1.3). While deliverable *D1.1 Survey of relevant low-resource languages* reports on the identification of monolingual and bilingual resources and deliverable *D1.3 Initial release of project data* provides pointer for downloading the corpora we have crawled and made available from the project webpage, this deliverable reports the work conducted on data gathering and data augmentation.

Crawling from the Internet of as much parallel data as possible and the application of data augmentation techniques is necessary for the development of neural machine translation (NMT) systems for under-resourced languages pairs, like those addressed in the GoURMET project. Crawling of parallel corpora is challenging when the amount of existing resources is extremely scarce, as it makes it difficult to identify reliable parallel segments. As regards data augmentation techniques, the standard technique in NMT, *backtranslation*, may fail if the initial systems are not good enough to boots translation performance through an iterative process.

The rest of this report is organised as follows. The next section reports our work on data crawling from the web, the problems encountered and the way the have been addressed. Section 3 describes the processing of the BBC and DW data dumps to obtain parallel corpora. Section 4 describes the strategy we have applied for generating synthetic training corpora and ongoing research. The report ends with a section dedicated to the work conducted within WP1 during the second half of the project and some concluding remarks.

2 Parallel and monolingual data crawling

This section describes the process followed to acquire parallel and monolingual data from the Internet as a resource to train MT systems. Deliverable *D1.3 initial release of project data* provides pointer for downloading them.

The process followed to acquire the corpora can be split into three different steps: crawling, document and segment alignment, and cleaning. The following is a description of the approaches followed in each of these steps.

2.1 Crawling

Two crawling approaches have been followed for this task. The first one consists of downloading as many documents as possible from a known collection of websites; the second one consists of exploring a specific top-level domain to find website from which to crawl data; the two approaches complement each other. For the first approach, we used `wget`[1] and `httrack`[2] to crawl a list of websites obtained by leveraging automatic-language-identification metadata from the CommonCrawl corpus:[3] websites containing at least 5 kB of text in the two targeted languages were crawled.

[1] https://www.gnu.org/software/wget/
[2] https://www.httrack.com/
[3] https://commoncrawl.github.io/cc-crawl-statistics/plots/languages

For the second one, we developed a specific tool, LinguaCrawl,[4] which has been released under a free/open-source license.

After crawling, the language of each document was identified using the CLD2 library,[5] then we extracted plain text from HTML/XML documents in the languages of interest, and performed sentence splitting.[6] As a results of this process we obtained *raw* monolingual data for the targeted languages in the crawl.

2.2 Document and segment alignment

In this step, parallel data is spotted from monolingual data in the two languages of interest. This stage is carried out using Bitextor (Espla-Gomis and Forcada, 2010),[7] a free/open-source tool to which we have contributed. First, candidate parallel documents are identified using a method based on bag-of-word-overlapping metrics. These metrics rely on bilingual lexicons, which we have automatically obtained by running `mgiza++` (Gao and Vogel, 2008) on the available parallel data.

Every pair of documents identified as parallel is then aligned at the sentence level using Hunalign (Varga et al., 2007). The same bilingual dictionary used for document alignment is provided to Hunalign in order to improve the accuracy of the alignment.

2.3 Cleaning

Cleaning corpora implies removing the noisy sentence pairs that are either incorrectly aligned or do not contain valid text in the expected language. This happens when a sentence is in a language different to that of the rest of the document or does not contain natural language.[8] First, language detection at the segment level is performed with CLD3,[9] then non-parallel sentence pairs are removed either using LASER[10] (Schwenk, 2018) or Bicleaner[11] (Sánchez-Cartagena et al., 2018), depending on the language pair (see below). The ready-to-use LASER models worked better than Bicleaner for the English–Turkish corpus, while Bicleaner performed better for the rest of corpora produced.

Bicleaner models are language-pair specific and use probabilistic bilingual dictionaries in both translation directions. These dictionaries were obtained as a by-product of the process of producing the bilingual lexicon for document alignment (see Section 2.2). The classifier used to score each sentence pair was trained on additional parallel data. In addition to the score provided by the classifier, a character-level language model was also used to provide monolingual confidence.

In order to decide whether to use LASER or Bicleaner and to find the appropriate threshold for the scores produced by each tool, an intentionally low threshold was set to generate a set of (likely

[4] https://github.com/transducens/linguacrawl
[5] https://github.com/CLD2Owners/cld2
[6] For document-level language identification, CLD2 was preferred to the newer CLD3 as it is supports language identification on HTML/XML directly, while CLD3 must be applied on plain text.
[7] https://github.com/bitextor/bitextor/branches
[8] For example, segments containing only numbers, punctuation marks or a list of tags in a blog.
[9] https://github.com/google/cld3/
[10] https://github.com/facebookresearch/LASER/
[11] https://github.com/bitextor/bicleaner/

Language pair	# sentences	# left tokens	# right tokens
English–Swahili	156,061	3,334,886	2,981,699
English–Turkish (Bicleaner)	308,303	5,735,675	4,853,602
English–Serbian	329,003	10,494,534	9,104,038
English–Serbo-Croatian	363,131	9,988,698	8,769,960
English–Amharic	57,835	1,001,376	782,378
English–Kyrgyz	14,497	248,401	165,895
Kyrgyz–Russian	23,016	285,427	304,356
English–Gujarati (PMIndia)	49,844	935,918	852,780
English–Tamil (PMIndia)	39,526	708,316	518,658

Table 1: Amount of parallel sentences and amount of left and right tokens in the corpora crawled from the Internet. For some language pairs we released different versions of these corpora or crawled additional corpora. The table only provides the data for one of these versions or corpora; additional information may be found in the corresponding section describing each corpus.

noisy) parallel sentences for their evaluation by the user partners. The result of this evaluation allowed us to decide, on a language-pair basis, whether to use Bicleaner or LASER and the threshold to be finally used in each case.

2.4 Crawled corpora

This section provides the specific details of the different corpora we have crawled. The usefulness of these parallel corpora for the project has been indirectly evaluated by measuring the quality of the NMT systems trained on them. Table 1 provides, for each corpus, the number of parallel sentences and tokens in each language. Deliverable *D1.3 Initial release of project data* provides pointers for downloading these corpora.

2.4.1 English–Swahili

A total of 3,751 websites were crawled for Swahili using `wget`. From the initial list of websites, 519 were not available at the time of crawling. Crawling was limited to 12 hours per website. The bilingual lexicon for document alignment was built on the concatenation of the following parallel corpora: EUBookshop v2,[12] Ubuntu,[13] and Tanzil.[14] A total of 180,520 pairs of documents were aligned, from which 2,051,678 segment pairs were extracted.

For the bicleaner model, the regressor was trained on the parallel corpus GlobalVoices2015[15] available at OPUS. The threshold for the score provided by the regressor was set to 0.68. Bicleaner's character-level language model was trained on the same corpora used to build the bilingual lexicons and the threshold was set to 0.5. The resulting parallel corpus consisted of 156,061 segment pairs.

[12] http://opus.nlpl.eu/EUbookshop-v2.php
[13] http://opus.nlpl.eu/Ubuntu-v14.10.php
[14] http://opus.nlpl.eu/Tanzil-v1.php
[15] http://casmacat.eu/corpus/global-voices.html

As a by-product of the crawling, a Swahili monolingual corpus containing 13,073,458 sentences was obtained. This corpus is the result of cleaning a larger one by applying automatic language detection and discarding sentences with less than 50% of alphanumeric characters.

We have published a paper (Sánchez-Martínez et al., 2020) which fully describes the preparation of this corpus and its use for training NMT systems.

2.4.2 English–Turkish

A total of 1,248 websites were crawled for Turkish using `wget`. From the initial list of websites, 5 were not available at the time of crawling. Crawling was limited to 12 hours per website. The bilingual lexicon for document alignment was built on the concatenation of the following parallel corpora: Bianet,[16] EUbookshop,[17] GlobalVoices,[15] KDE4,[18] OpenSubtitles 2018,[19] Tanzil,[14] Tatoeba,[20] Ubuntu,[13] and Wikipedia.[21] A total of 110,399 pairs of documents were aligned, from which 4,740,723 segment pairs were extracted.

The regressor of Bicleaner was trained on the concatenation of the parallel corpora GlobalVoices,[15] Tanzil,[14] and GNOME.[22] The threshold for the score provided by the regressor was set to 0.59. Bicleaner's character-level language model was trained on the same corpora used to build the bilingual lexicons and the threshold was set to 0.5. The resulting parallel corpus consisted of 308,303 pairs of segments.

For Turkish–English, additional corpora were produced using LASER. LASER allows to rank segment alignments according a score related to the similarity of the multilingual embeddings of the segments. In this case, three different corpora were produced: one that is focused on precision (extremely low amount of noise in the corpus) with 200,000 pairs of segments, another focused on recall (the corpus is not too noisy and most of the parallel data has been identified) with 600,000 pairs of segments, and an intermediate corpus with about 400,000 segment pairs.

As a by-product of the crawling, a Turkish monolingual corpus containing 20,158,159 sentences was obtained. This corpus is the result of cleaning a larger one. In addition to language detection with CLD3, sentences with less than 50% of alphanumeric characters were removed.

2.4.3 English–Amharic

A total of 3,378 websites were crawled for Amharic using `wget`. From the initial list of websites, 435 were not available at the time of crawling. Crawling was limited to 12 hours per website. The bilingual lexicon for document alignment was built on the concatenation of the following parallel corpora: Wikimedia,[23] Tatoeba,[20] JW300,[24] and bible-uedin v1.[25] A total of 49,826 pairs of documents were aligned, from which 1,090,636 segment pairs were extracted.

[16] http://opus.nlpl.eu/Bianet.php
[17] http://opus.nlpl.eu/EUbookshop.php
[18] http://opus.nlpl.eu/KDE4.php
[19] http://opus.nlpl.eu/OpenSubtitles2018.php
[20] http://opus.nlpl.eu/Tatoeba-v2.php
[21] http://opus.nlpl.eu/Wikipedia-v1.0.php
[22] http://opus.nlpl.eu/GNOME-v1.php
[23] http://opus.nlpl.eu/wikimedia-v20190628.php
[24] http://opus.nlpl.eu/JW300-v1.php
[25] http://opus.nlpl.eu/bible-uedin-v1.php

For the bicleaner model, the regressor was trained on the concatenation of the parallel corpora GlobalVoices,[15] Tanzil,[14] GNOME,[22] and Ubuntu.[13] The threshold for the score provided by the regressor was set to 0.60. Bicleaner's character-level language model was trained on the same corpora used to build the bilingual lexicons and the threshold was set to 0.1. The resulting parallel corpus consisted of 57,835 segment pairs.

As a by-product of the crawling, a Amharic monolingual corpus containing 3,265,235 sentences was obtained. This corpus is the result of cleaning a larger one. In addition to language detection with CLD3, sentences with less than 50% of alphanumeric characters were removed.

2.4.4 English–Kyrgyz and Kyrgyz–Russian

Kyrgyz is the most under-resourced language addressed to date. For this reason, additional effort was put to obtain data for it, namely, by applying two different crawling strategies, as described in Section 2.1: the use of `wget` on a collection of seed URLs and, the development and use of LinguaCrawl to find new websites containing more parallel data from the `.kg` top level domain. The initial list of seed websites consisted of 18 hosts. This list was manually extracted from a slightly larger list obtained by using the method based on preliminary language identification on CommonCrawl. Given the reduced amount of websites to be crawled, crawling was limited to 3 days for each one. In order to make the most from the small amount of websites crawled, not only Kyrgyz–English parallel data were aligned, but also Kyrgyz–Russian.

The bilingual lexicons for document alignment were built on the concatenation of the following parallel corpora: Wikimedia,[23] QUED,[26] and JW300.[24] A total of 3,202 pairs of documents were aligned for Kyrgyz–English, from which 61,800 segment pairs were extracted. For Kyrgyz–Russian, 21,623 document alignments were obtained, and 534,869 pairs of segments were aligned.

The Bicleaner model for regression was trained on the concatenation of the parallel corpora GNOME,[22] Tatoeba,[20] and Ubuntu.[13] The threshold for the score provided by the regressor was set to 0.5. Bicleaner's character-level language model was trained on the same corpora used to build the bilingual lexicons and the threshold was set to 0.5. The resulting parallel corpora consisted of 14,497 and 23,016 segment pairs for Kyrgyz–English and Kyrgyz–Russian, respectively.

As a by-product of the crawling, a Kyrgyz monolingual corpus containing 845,913 sentences was obtained. This corpus is the result of cleaning a larger one. In addition to language detection with CLD3, sentences with less than 50% of alphanumeric characters were removed, and the same language model trained for cleaning parallel data was also applied to the monolingual corpus.

2.4.5 English–Gujarati and English–Tamil

We started our parallel crawling for English–Gujarati by selecting 2,486 domains containing both Gujarati and English text, according to the language-tagged CommonCrawl (Buck et al., 2014), and crawling using the `httrack` crawler. Early efforts to extract a corpus were unsuccessful because many of the domains did not, in fact, contain Gujarati text, there were issues with the support of Indian scripts in the tool-chain, and document and sentence alignment was inaccurate due to the lack of existing parallel resources for bootstrapping.

[26] http://opus.nlpl.eu/QED.php

Examining the domains crawled, we realised that those belonging to the Indian government (i.e. with URLs in the `gov.in` domain) offered the best possibility for extraction of parallel data: when these sites contained English and Gujarati, the text was of good quality and a brief inspection suggested that they contained parallel text. There were 171 `gov.in` domains in the crawl. Using the Pavlick dictionaries (Pavlick et al., 2014) as an additional resource during sentence alignment, we extracted an initial corpus of 171,943 sentence pairs. Since there was little parallel data available for training a bicleaner model, and no LASER model was available for Gujarati, we applied some simple length heuristics and language identification for cleaning, resulting in a corpus of 10,650 sentence pairs. This corpus was released as the "govin" corpus for the WMT19 news translation task (Barrault et al., 2019).

After creating the Gujarati–English corpus, we performed a wider crawl of all `gov.in` domains listed in the CommonCrawl data releases, to look for parallel corpora in other languages of India. On examining this crawl, we noted that, for many languages, by far the most productive site for parallel data was the website of the India prime minister (http://www.pmindia.gov.in).

Given that the PMIndia website contained many articles written in English, and in up to 13 of the major languages of India, we decided to focus our efforts on extracting parallel corpora from this site. In order to crawl the articles, we developed a custom crawler, since the structure of the site meant that the Bitextor was not able to access all the articles. In all we obtained between 1,413 and 5,722 articles for each language. Text extraction was straightforward using Alcazar[27] but in order to sentence split the text for alignment we had to update the Moses splitter (Koehn et al., 2007) to handle the various scripts used in India, and to better handle the many itemised lists in this corpus.

For sentence alignment, we experimented with two methods: Hunalign (Varga et al., 2005), with a Pavlick dictionary where available; and Vecalign (Thompson and Koehn, 2019), a method based on sentence embeddings. Evaluation showed that both methods performed similarly, with about 88% of a sample of Tamil–English sentence pairs judged to be aligned correctly. Where possible (i.e. where sentence embeddings were available) we took the intersection of both alignment methods for the released corpus.

The final corpus contains between 56,831 (for Hindi–English) and 7,484 (for Manipuri–English) sentence pairs, depending on the language. We have published a paper (Haddow and Kirefu, 2020) which fully describes the preparation and evaluation of this corpus.

2.4.6 English–Serbian and English–Serbo-Croatian

Two separate corpora were created for Serbian from the data crawled from the web: one containing only English–Serbian sentence pairs, and another containing sentences in Serbian, Croatian or Bosnian on one side and English on the other. The reason for allowing all three languages is that due to their similarity, language identification can be unreliable. The corpus containing only Serbian was crawled from websites verified to contain pages in Serbian and English and not the other two languages, while the corpus allowing Serbian, Bosnian and Croatian was crawled from a list of websites relying on automatic language identification.

For the English–Serbo-Croatian corpus we used the automatic-language-identification metadata from the Common-Crawl corpus (Buck et al., 2014) to select websites that contain at least 100 kB of text in either one of Serbian, Croatian and Bosnian, as well as English, resulting in a total of 7,876 websites. The maximum crawl time for each website was set to 12 hours. We crawled the

[27] https://github.com/saintamh/alcazar

websites using `wget`. For the English–Serbian corpus we manually compiled a list of 208 websites that were confirmed to contain comparable pages in Serbian and English. The maximum crawl time was set to one week. We crawled the websites using `wget`.

For document alignment in both cases we used Bitextor's option to use an external MT system to translate source to target text and compute a TF/IDF score to match documents in the two languages. We used an existing Serbian→English system to translate documents in Serbian/Bosnian/Croatian into English. The threshold for document alignment was set to 0.1.

Segment alignment was done using the translations produced for document alignment and the Bitextor option to use Bleualign [28]. The Bleualign threshold was set to 0.1. The alignment produced 476,022 sentence pairs for the English–Serbian corpus and 662,918 for the English–Serbo-Croatian corpus.

We used Bicleaner to clean the resulting parallel corpora, using an officially released English–Croatian Bicleaner model. All Bosnian and Serbian data was transliterated into Latin if originally written in Cyrillic. We filtered out sentences with a Bicleaner score lower than 0.6. This resulted in 348,639 sentence pairs in the English–Serbian corpus and 432,572 in the English–Serbo-Croatian corpus corpus.

We applied Bifixer[29] to the cleaned parallel sentences and de-duplicated both corpora. Sentence pairs that appear in the English–Serbian corpus were removed from the English–Serbo-Croatian corpus to avoid a large overlap between them. The final corpora contain 329,003 sentence pairs in the English–Serbian corpus and 363,131 sentence pairs in the English–Serbo-Croatian corpus.

2.5 Monolingual News Crawl

The University of Edinburgh has been releasing monolingual news crawls annually for many years for the languages covered in the WMT shared tasks. For GoURMET, we have extended these releases to cover all the non-English languages considered in the project so far, plus several Indian languages. We have also added several African language news sources to the crawl and will be releasing corpora for these too. The news crawl is available at http://data.statmt.org/news-crawl.

The operation of the news crawl is straightforward. We maintain a list of RSS feeds for news sources around the world, and these are checked daily for new stories. We then download and store the HTML version of all new stories. To extract text, we use Alcazar, then we sentence-split using Moses tools, de-duplicate and shuffle.

In order to add new sources for a new language, we used to have to find online news sites for that language manually, then search for RSS feeds (which are often not linked directly from the front page). This could be quite a slow process, especially with unfamiliar languages and scripts. In order to automate the process of adding new feeds, we exploited the directory of online news sources at http://www.abyznewslinks.com/. We crawled this site to create a database of news sources, along with their language and URL, then we created another crawler to examine each site in this database, looking for RSS feeds. This process allows us to quickly identify feeds for any given language.

The latest news crawl release (January 2020) contains news corpora for the 8 non-English GoURMET languages considered so far, plus several languages of India (Bengali, Hindi, Kannada,

[28] https://github.com/rsennrich/Bleualign
[29] https://github.com/bitextor/bifixer

Language	# documents BBC	# documents DW
Amharic	4,808	18,316
English	236,860	178,463
Kyrgyz	5,000	—
Serbian	4,948	49,092
Tamil	20,302	—

Table 2: Number of documents in the monolingual data dumps provided by BBC and DW for the second round of languages.

Malayalam, Marathi, Odia, Punjabi, Tamil and Telugu). In total the release contains 220 GB of compressed corpora, covering 41 languages, with 1.17B sentences.

3 Processing of BBC and DW data dumps

In addition to the crawling of corpora, we have processed data dumps provided by the user partners for the second round of languages. These data dumps consist of collections of pieces of news that have been published by BBC or DW in their respective websites. They have been processed mainly to obtain test corpora, except for Serbian for which we also obtained data for training. It is worth noting that to be able to freely distribute our NMT models, BBC data cannot be used for training, since BBC has restrictions on derivative work. Table 2 reports the number of documents in these data dumps.

3.1 BBC data dumps

BBC data dumps were provided in JSONL[30] format including, not only the body of each piece of news, but also additional information such as the headline, the URL where it was published, and the identifiers of the images included, among others.

As regards the relation between the documents across the monolingual data dumps, it is worth noting that for a piece of news in the English data dump:

- there may be not be a translated version in another monolingual data dump;
- there may be pieces of news on the same topic in other languages that are not the translation of the English piece of news, and
- there may be pieces of news in other languages that are only partial translations, i.e. the original piece of news in English has been reversioned into another language and the resulting piece of news does not provide the same information or it is structured in a different way.

Data dumps can therefore be considered comparable corpora, consisting of comparable documents. In order to obtain from them parallel segments to be used for testing we had to adapt the process described in Section 2 because the tools and methods used for parallel data demonstrated not to be valid to get high quality parallel segment from these data dumps. The reminder of this section explains these adaptations.

[30] http://jsonlines.org/

3.1.1 Document alignment

The first step to process the data dumps was to identify which pairs of documents could contain parallel data; this process was needed because BBC did not have the document alignments. To do so, we relied on the image identifiers used in each piece of news; as it revealed to be the best approach in our preliminary experiments because they are language independent. A metric based on the inverse document frequency (IDF) was used for this. Namely, for every pair of documents between two monolingual data dumps, the following metric was computed:

$$\text{Score}(D_1, D_2) = \frac{1}{|\{\text{Img}(D_1) \cap \text{Img}(D_2)\}|} \sum_{i \in \{\text{Img}(D_1) \cup \text{Img}(D_2)\}} \frac{1}{\text{NumDocs}(i)}$$

where $\text{Img}(D)$ returns the set of image ids used in document D and $\text{NumDocs}(i)$ the number of documents where image i is used. This metric scores higher the pairs of documents sharing as many images as possible, but also gives a higher weight to those that are infrequent. After an initial human validation on the English–Amharic preliminary results, a threshold was set so that only document alignments with a score equal or higher to 0.1 were kept.

3.1.2 Segment alignment

Tools such as Hunalign are designed to align segments in documents that are parallel, which, in general, is not the case of aligned documents from the BBC data dumps. LASER was considered to be used for this task, but preliminar experiments reported poor quality for several languages in this project, such as Kyrgyz and Amharic. For this reason, we considered all possible segment pairs for each pair of aligned document (the Cartesian product of them), ranked them as described next, and selected only the best ones.

3.1.3 Segment-pair ranking

After several unfruitful attempts with Bicleaner —we tried with a new noise model and new features that improve it but not to the point of being useful to have a corpus useful for testing—, we decided to use Google Translate, as it is available for all the languages in Table 2, and an automatic evaluation metric as described next.

Every segment in a language different from English was translated into English using Google Translate, then the chrF2++ metric between the MT output and the English side of the pair was computed. An inverse ranking was produced for each pair of languages, and the 3 000 most-promising segment pairs with more than 5 words in each side were kept for their validation and use as test sets.

3.1.4 Human curation

A form with the 3 000 most-promising segment pairs was provided to human validators so that they could confirm which segment pairs were actually parallel in order to used them for testing. Validators were asked to manually check and mark parallel segment pairs until a total of 1 000 parallel pairs were identified; those pairs of segments in the higher positions of the ranking are likelier to be parallel. By the time of writing these lines this human curation was finished for Tamil and Serbian and Kyrgyz.

3.2 DW data dump

We crawled the DW website using their web API. To do this we first obtained a list of all articles in the languages of interest by parsing the sitemap.[31] For every language, there are a number of sitemap details, each for a different time period. These contain information about articles published within this time period in the relevant language, including the URL, from which we obtained the article IDs, which were necessary for crawling. The crawled articles were stored in a JSONL format, with fields containing the content of the article as well as metadata such as original publishing date, article ID, headline and others. For the extraction of parallel English–Amharic data, the JSONL DW data also contained image IDs. Unfortunately these image ids did not allow us to obtain reliable document alignments to proceed as we did with the BBC data dumps. As a result we only processed the English and Serbian data dumps to obtain additional data for training.

3.2.1 English–Serbian alignment

For English–Serbian, alignment was done using a modified version of Bitextor (Espla-Gomis and Forcada, 2010) which uses the publishing dates of the documents as an additional filter for document alignment. Documents were aligned using the option for an external MT system, translating the Serbian documents into English and using TF/IDF similarity to align them. The alignment was then filtered, restricting the final aligned documents to have been published within 30 days of each other.

Segment alignment was done using the translations from document alignment and Bleualign.[32] The resulting sentence pairs were cleaned using Bicleaner, using an official English–Croatian Bicleaner model, and a threshold of 0.5. This was possible since Serbian and Croatian are very similar. We applied Bifixer[33] to the final corpus, resulting in a total of 90,825 sentence pairs. Of these, a dev and a test set were selected, containing 2,100 sentence pairs each, and 86,623 were left for training (two sentences were removed due to duplication with the dev/test sets).

The unaligned Serbian and English sentences were used to produce monolingual corpora. The Serbian monolingual corpus contains 1,106,400 sentences and the English monolingual corpus contains 4,161,101 sentences.

4 Data augmentation

In order to automatically generate synthetic parallel corpora we have applied the back-translation approach proposed by Sennrich et al. (2016); in one case the system used for back translation was a statistical machine translation system, in the rest of cases it was a NMT system. In addition, for some language pairs we applied an iterative back-translation algorithm (Hoang et al., 2018) that simultaneously leverages monolingual data in the two languages involved in the translation. See Section 3 in deliverable *D5.3 Initial integration report* for the details of the data augmentation techniques used for building each translation model.

The next subsection explains ongoing research on the use of variational inference with neural networks for generating synthetic data.

[31] https://www.dw.com/sitemap.xml
[32] https://github.com/rsennrich/Bleualign
[33] https://github.com/bitextor/bifixer

4.1 Variational inference for the generation of synthetic data

NMT systems reach their peak performance when there are large amounts of parallel data available for the addressed language pair. Representations of words that do not appear in diverse contexts are usually poorly estimated (Fadaee et al., 2017), and this problem can be exacerbated in low-resource scenarios.

UA and UvA have jointly explored the use of variational autoencoders (Bowman et al., 2016) that operate on sentences to generate synthetic text with the aim of enhancing the diversity of an existing dataset and improving the performance of the NMT systems trained on it. Variational autoencoders represent sentences as continuous distributions in a smooth latent space. That means that one can, in theory, generate new sentences from any point in the space, and also interpolate between existing sentences by generating from points along a path connecting their distributions in latent space. This has the potential to lead to sentences that are considerably different from those in the training data.

VAEs suffer from the well-known posterior collapse problem (Alemi et al., 2018). When a strong generator is employed, such as an autoregressive language model in the sentence VAEs used for data augmentation, the model tends to ignore the latent variable. Various strategies have been proposed in the literature for fighting posterior collapse, including work developed in WP3 (Pelsmaeker and Aziz, 2020). In addition to the well-established *free bits* method (Kingma et al., 2016), the inclusion of auxiliary tasks that operate on different views of the data has been explored (Zhao et al., 2017). These views are designed so that they enforce the latent variable to contain information that is useful for the main decoder. Some of the tasks explored are:

- Bag-of-words vocabulary prediction: prediction of the vocabulary of each sentence, conditioned only on the latent variable, by means of a feed forward neural network.
- Autoregressive vocabulary prediction: prediction of the vocabulary of each sentence, conditioned on the latent variable and partial evidence about the sentence vocabulary itself by means of a masked autoencoder (Germain et al., 2015).
- Autoregressive vocabulary and frequency prediction: prediction of the words in the sentence (words that appear multiple times are predicted multiple times) in an autoregressive way by means of a recurrent neural network language model that operates on a random permutation of the words in the sentence.

These auxiliary tasks were introduced in the training objective by means of constrained optimisation (Boyd et al., 2004). Systems were trained so as to optimise the Evidence Lower Bound (ELBO) objective (Jordan et al., 1999) subject to the expected log-likelihood of the auxiliary task being above certain rate t, as shown the following equation:

$$\max_{\theta,\lambda,\phi} \; \mathbb{E}_X \left[\text{ELBO}(\theta, \lambda|x)\right]$$
$$\text{s.t. } \mathbb{E}_X \left[E_{Z|X,\lambda} \left[\log p_{X|Z}(x|z,\phi)\right]\right] > t$$

where X is a random variable that represents the observable data points, Z is the latent variable, and λ, θ and ϕ parameterise respectively the inference network, generative main decoder and generative auxiliary decoder. Constrained optimisation was approximated by means of a Lagrangian

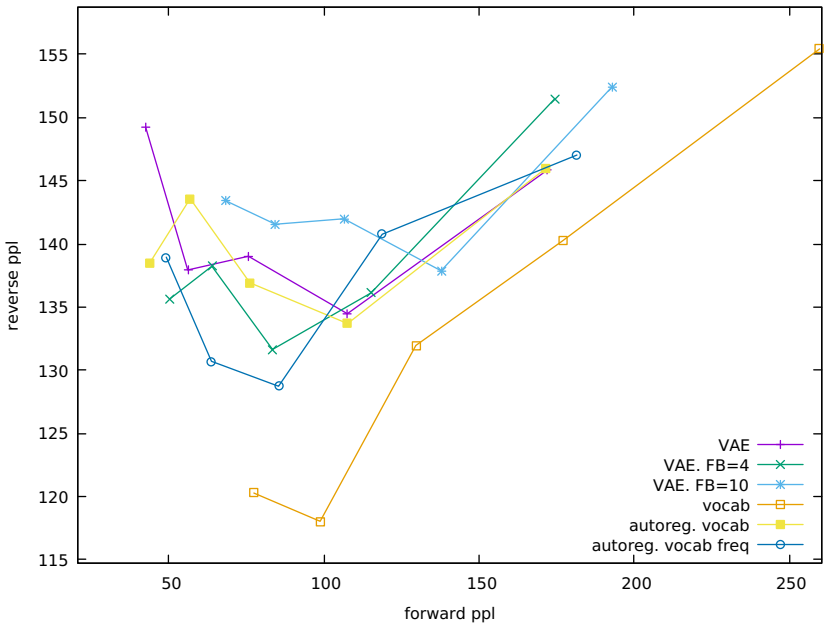

Figure 1: Forward and reverse perplexities for values of nucleus sampling between 0.5 and 1.0.

multiplier, in a similar way to the Minimum Desired Rate method developed in WP3 (Pelsmaeker and Aziz, 2020).

VAEs were trained with different values of free bits and with the three proposed auxiliary tasks on the English News Commentary v12 corpus. None of the auxiliary tasks was able to outperform the free bits method in terms of perplexity on a held-out test set. However, a text generation evaluation (Caccia et al., 2018) showed a different picture. Figure 1 shows the forward and reverse perplexities of a sample of text obtained with the different VAEs for different values of nucleus sampling p (Holtzman et al., 2019). According to Caccia et al. (2018), this representation allows us to evaluate the quality/diversity trade-off of text generation models. The plot shows that the VAE with the the bag-of-words auxiliary task outperforms the other methods in terms of this quality/diversity ratio: for similar forward perplexity (that accounts for fluency of the generated text), it achieves lower reverse perplexity (that accounts for both fluency and diversity). This result makes the text generated with this VAE suitable for data augmentation, as it would provide additional contexts for more words.

Once the bag-of-words auxiliary task was identified as an appropriate method for generating diverse and fluent text with a VAE, data augmentation was carried out as follows. Given an out-of-domain parallel corpus and a target-language in-domain monolingual corpus, a sentence VAE was trained on the monolingual corpus. Additional synthetic monolingual data was generated via ancestral sampling: sampling from the latent prior (a multivariate normal distribution) and then sampling from the language model categorical distribution at each time step. In order to avoid generating low-probability words from the categorical distribution, nucleus sampling (Holtzman et al., 2019) was used. Finally, both the original monolingual corpus and the synthetic one were backtranslated and concatenated to the parallel corpus.

Preliminary results on the German-to-English language pair using TED talks as parallel data and news fragments of similar size as monolingual data show moderate improvements (around 1 BLEU point) on WMT test sets.

5 Future work

During the second half of the project we plan to work on the following things to improve the outcomes of this workpackage:

- Continue crawling corpora from the web to provide to the academic partners corpora for the development of NMT systems for the next rounds of languages.
- Research on methods for improving the alignment and cleaning of the corpora crawled from the Internet. Current methods rely on bilingual resources that for some language pairs are really scarce and this makes them to be of little use. We plan to study different ways of improving the results obtained with Bicleaner and LASER. As regards Bicleaner, it currently relies on the use of probabilistic bilingual dictionaries automatically learned from existing parallel data. We plan to improve Bicleaner by using monolingual data, which is easier to find for under-resourced languages, by using word embeddings to compare sentences in the two languages, in a way similar to that of Bernier-Colborne and Lo (2019). As regards LASER, we will study which are the most successful ways of adding support (or training new models) for languages that are not currently supported. We plan to apply and evaluate the methods we will develop on both the parallel data obtained through crawling and the data dumps provided by the user partners.
- In the data augmentation with variational autoencoders research line, we plan to carefully analyse the properties of the text sampled from the system that included the bag of words auxiliary task to get a deeper insight about the properties that make it more suitable for data augmentation than the rest of alternatives. We also consider devising new auxiliary tasks and study possible ways of discarding synthetic sentences not containing relevant new information.
- We are pursuing work augmenting parallel training data by creating synthetic examples of sentences containing rarely seen words. We are extending the approach of Fadaee et al. (2017) by generating sentence pairs containing rare words in new contexts. We will be selecting these contexts from existing parallel corpora, by finding sentences with similar contexts according to a large monolingual pre-trained language model and then substituting the rare word in this context.

6 Conclusion

This deliverable has reported the work conducted within WP1 on data gathering and data augmentation. We crawled data from the web to obtained training resources; we faced some difficulties due to the scarceness of bilingual resource to be used to identify additional parallel corpora. We have also processed data dumps provided by the user partners to obtain additional training resources and quality in-domain test corpora. Finally we are working on new methods to generate synthetic training data for their use with state-of-the-art methods for data augmentation such as back-translation.

A summary of the research outcomes of WP1 follows:

Corpora. We have released all the corpora we have crawled from the web. Deliverable *D1.3 Initial release of project data* provides the URLs from which they can be downloaded.

Publications. The following research papers resulted from the work described in this deliverable:

- *PMIndia — A Collection of Parallel Corpora of Languages of India*, Haddow and Kirefu (2020)
- *An English-Swahili parallel corpus and its use for neural machine translation in the news domain*, Sánchez-Martínez et al. (2020)

Software. The following is a list of free/open-source software we have released or contributed to:

- LinguaCrawl (https://github.com/transducens/linguacrawl). Developed within WP1. Tool that allows to crawl a number of top-level domains to download any text documents in the languages specified by the user.
- LASER train (https://github.com/transducens/LASERtrain). Developed within WP1. This piece of software reproduces the architecture of Artetxe and Schwenk (2018, 2019) to train language-agnostic sentence embeddings. Artetxe and Schwenk (2018, 2019) released a large model covering 93 languages as part of the LASER project but did not release the code used to train them.
- Bitextor (https://github.com/bitextor/bitextor). Contributed to. It is the most widely-used tool to automatically harvest bilingual corpora from multilingual websites.
- Bicleaner (https://github.com/bitextor/bicleaner). Contributed to. It is a tool for the detection of noisy sentence pairs in a parallel corpus.

References

Alemi, A., Poole, B., Fischer, I., Dillon, J., Saurous, R. A., and Murphy, K. (2018). Fixing a Broken ELBO. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 159–168, Stockholmsmässan, Stockholm Sweden. PMLR.

Artetxe, M. and Schwenk, H. (2018). Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. arXiv:1812.10464.

Artetxe, M. and Schwenk, H. (2019). Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy. Association for Computational Linguistics.

Barrault, L., Bojar, O., Costa-jussà, M. R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Koehn, P., Malmasi, S., Monz, C., Müller, M., Pal, S., Post, M., and Zampieri, M. (2019). Findings of the 2019 conference on machine translation (wmt19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

Bernier-Colborne, G. and Lo, C.-k. (2019). NRC parallel corpus filtering system for WMT 2019. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 252–260, Florence, Italy. Association for Computational Linguistics.

Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A., Jozefowicz, R., and Bengio, S. (2016). Generating sentences from a continuous space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany. Association for Computational Linguistics.

Boyd, S., Boyd, S. P., and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

Buck, C., Heafield, K., and van Ooyen, B. (2014). N-gram counts and language models from the common crawl. In *Proceedings of the Language Resources and Evaluation Conference*.

Caccia, M., Caccia, L., Fedus, W., Larochelle, H., Pineau, J., and Charlin, L. (2018). Language gans falling short.

Espla-Gomis, M. and Forcada, M. (2010). Combining content-based and url-based heuristics to harvest aligned bitexts from multilingual sites with bitextor. *The Prague Bulletin of Mathematical Linguistics*, 93(1):77–86.

Fadaee, M., Bisazza, A., and Monz, C. (2017). Data augmentation for low-resource neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 567–573, Vancouver, Canada. Association for Computational Linguistics.

Gao, Q. and Vogel, S. (2008). Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57, Columbus, Ohio, USA.

Germain, M., Gregor, K., Murray, I., and Larochelle, H. (2015). Made: Masked autoencoder for distribution estimation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 881–889. JMLR.org.

Haddow, B. and Kirefu, F. (2020). PMIndia – A Collection of Parallel Corpora of Languages of India. *arXiv e-prints*, page arXiv:2001.09907.

Hoang, V., Koehn, P., Haffari, G., and Cohn, T. (2018). Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia.

Holtzman, A., Buys, J., Forbes, M., and Choi, Y. (2019). The curious case of neural text degeneration. *CoRR*, abs/1904.09751.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An Introduction to Variational Methods for Graphical Models. *Machine Learning*, 37(2):183–233.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. (2016). Improved variational inference with inverse autoregressive flow. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29*, pages 4743–4751. Curran Associates, Inc.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the Association for Computational Linguistics Companion Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Pavlick, E., Post, M., Irvine, A., Kachaev, D., and Callison-Burch, C. (2014). The language demographics of Amazon Mechanical Turk. *Transactions of the Association for Computational Linguistics*, 2(Feb):79–92.

Pelsmaeker, T. and Aziz, W. (2020). Effective estimation of deep generative language models. In *ACL*.

Sánchez-Cartagena, V. M., Bañón, M., Ortiz-Rojas, S., and Ramírez-Sánchez, G. (2018). Prompsit's submission to WMT 2018 parallel corpus filtering shared task. In *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*, Brussels, Belgium.

Schwenk, H. (2018). Filtering and mining parallel data in a joint multilingual space. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 228–234, Melbourne, Australia. Association for Computational Linguistics.

Sennrich, R., Haddow, B., and Birch, A. (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, Germany.

Sánchez-Martínez, F., Sánchez-Cartagena, V. M., Pérez-Ortiz, J. A., Forcada, M. L., Esplà-Gomis, M., Secker, A., Coleman, S., and Wall, J. (2020). An English–Swahili parallel corpus and its use

for neural machine translation in the news domain. In *Proceedings of the 22th Annual Conference of the European Association for Machine Translation*, pages 299–308, Online Conferece. European Association for Machine Translation.

Thompson, B. and Koehn, P. (2019). Vecalign: Improved sentence alignment in linear time and space. 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 1342–1348, Hong Kong, China. Association for Computational Linguistics.

Varga, D., Halácsy, P., Kornai, A., Nagy, V., Németh, L., and Trón, V. (2007). Parallel corpora for medium density languages. *Amsterdam Studies In The Theory And History Of Linguistic Science Series 4*, 292:247.

Varga, D., Németh, L., Halácsy, P., Kornai, A., Trón, V., and Nagy, V. (2005). Parallel corpora for medium density languages. In *Proceedings of RANLP*.

Zhao, T., Zhao, R., and Eskenazi, M. (2017). Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–664, Vancouver, Canada. Association for Computational Linguistics.

ENDPAGE

GoURMET

H2020-ICT-2018-2 825299

D1.2 Initial progress report on data gathering and augmentation