Lingua Custodia @ Covid-19 MLIA - Eval Initiative

Raheel Qader

Lingua Custodia , Paris, France

Abstract. This paper describes the participation of Lingua Custodia in the first two rounds of the Covid-19 MLIA @ Eval Initiative. We compare bilingual and multilingual machine translation models to translate English to 7 languages including French, German, Spanish, Italian, Greek, Swedish and Arabic. The results show that our models outperform other participating systems in several language pairs.

1 Introduction

The wide-spread of covid-19 has caused major health and economic problems around the world. The sudden appearance of this virus has lead to difficulties in communication between nations as most current Machine Translation (MT) engines do not recognize covid-19 related terminology, and thus, not able to properly translate such text. The idea of the Covid-19 MLIA - Eval Initiative is to accelerate the creation of necessary resources and tools in order to improve the quality of current MT systems in the context of covid-19 [3].

The initiative is basically a challenge of 3 rounds. At each round, the organizers release training, development, and test sets and participants have to develop MT models using this data only (called constrained MT) or they can opt to use additional data (called unconstrained MT). The first round of the evaluation initiative addresses 6 language pairs: English to German, English to French, English to Spanish, English to Italian, English to Modern Greek and English to Swedish, while in the second round, Arabic has been added as well, making the total number of language pairs 7.

This paper describes the participation of Lingua Custodia in this initiative. In the first round, we participate in all but the English to Modern Greek task, and in the second round, all language pairs. Since the source language is always limited to English, we experiment with multilingual machine translation approach as well as bilingual models. In both rounds, we only participate in the constrained translation task.

The rest of the paper describes the data processing, the proposed MT architecture and the conducted experiments for the first and second rounds of the challenge.

Covid-19 MLIA @ Eval Initiative, `http://eval.covid19-mlia.eu/`.

2 Round 1

2.1 Data

This section gives details of the data provided by organizers of the challenge and the pre-processing steps done by our team in order to prepare the data for training the 5 engines.

As stated earlier, in the first round we only participate in the English to German, French, Spanish, Italian and Swedish tasks. Table 1 shows the statistics of the data used for the six language directions. The French and Spanish directions have the largest training data with almost one million sentences each while the Greek and Swedish direction have the fewest. The validation sets vary significantly from one direction to another, with the German having the smallest set (528 sentences) and the Greek direction having the largest set (3.9K sentences). All language directions have a test set of 2K sentences.

Table 1. Corpora statistics. $|S|$ stands for number of sentences, $|T|$ for number of tokens and $|V|$ for size of the vocabulary. M denotes millions and K thousands.

		German		**French**		**Spanish**		**Italian**		**Modern Greek**		**Swedish**	
		En	**De**	**En**	**Fr**	**En**	**Es**	**En**	**It**	**En**	**El**	**En**	**Sv**
Train	$\|S\|$	927K		1M		1M		900K		834K		807K	
	$\|T\|$	17.3M	16.1M	19.4M	22.6M	19.5M	22.3M	16.7M	18.2M	15.0M	16.4M	14.5M	13.2M
	$\|V\|$	372.2K	581.6K	401.0K	438.9K	404.4K	458.0K	347.7K	416.0K	305.7K	407.5K	298.2K	452.0K
Validation	$\|S\|$	528		728		2.5K		3.7K		3.9K		723	
	$\|T\|$	8.2K	7.6K	17.0K	18.8K	48.9K	56.2K	78.2K	84.0K	73.0K	72.7K	11.4K	10.0K
	$\|V\|$	2.4K	2.6K	4.1K	4.5K	9.7K	10.6K	12.4K	14.9K	10.3K	14.5K	2.6K	2.8K
Test	$\|S\|$	2000		2000		2000		2000		2000		2000	
	$\|T\|$	34.9K	33.2K	33.2K	35.8K	32.6K	34.3K	33.7K	34.2K	42.6K	44.3K	35.3K	30.6K
	$\|V\|$	7.8K	9.6K	6.7K	7.7K	6.7K	7.9K	8.6K	10.4K	9.5K	12.5K	7.1K	8.2K

2.2 Machine translation models

We experimented with several techniques to train our models including training models for a specific language pair, and models that can translate from English to several languages. In addition to that, we performed few steps of pre-possessing on the provided data.

Pre-processing In order to prepare the data for training, we first clean the training data of all language pairs by removing very long sentences. Then, instead of performing a specific tokenization step (e.g., using Moses [5]), we applied SentencePiece [6] for subword segmentation, which applies tokenization as well. We force sentencepiece to split numbers character-by-character. This will reduce the total number of digit combinations (i.e., 10) that the model has to see, thus,

Table 2. Results on the constrained machine translation task. Systems are scored by BLEU and chrF.

	En-De		**En-Fr**		**En-Es**		**En-It**		**En-Sv**	
	BLEU	**chrF**	**BLEU**	**chrF**	**BLEU**	**chrF**	**BLEU**	**chrF**	**BLEU**	**chrF**
Single model	26.7	0.556	48.9	0.703	-	-	-	-	-	-
Multilingual	29.5	0.584	49.0	0.705	47.6	0.698	28.4	0.572	30.4	0.589

it can generalize much easier on numbers. We use a unigram SentencePiece by creating a shared vocabulary between source and target sequences with 50K for bilingual and 70K for multilingual models.

Model architectures Our first round machine translation models are all based on the transformer architecture [9]. We use the Seq2SeqPy toolkit [7], which is a very lightweight toolkit with several sequence-to-sequence implementations including the transformer model.

Bilingual models:
Bilingual models are basically models that take a sequence in the source language and translates it to a sequence in the target language. We need to train as many models as the number of language directions with such an architecture.
Multilingual models: on the contrary, a multilingual model can be trained such that one single model can translate between several language directions. Multilingual machine translation can be implemented in several ways. One approach is to add a token in the beginning of the source sequence in order to indicate the target language (e.g., 2fr, 2de, 2es). Another approach is to use source factors, i.e., to attach the embedding of language-specific id to the embedding of each token in the source sequence. In our experiments, for the sake of simplicity, we use the former approach.
Hyper-parameters
For both model types we use the standard transformer architecture with 6 encoder and 6 decoder layers. The size of the embedding and hidden states are set to 512 while the size of the feed-forward layer is 2048 and we use 8 attention heads. The source and target embeddings are tied with the vocabulary projection layer. The batch size is set to 80 and source/target max lengths were capped at 120. We use Adam optimizer with learning rate of 0.0002, a warmup step of 5000, and label smoothing of 0.1. Finally, during inference, we use a beam size of 5. Our models are trained on 5 RTX 2080 Ti gpus.

2.3 Experiments

In this section we describe the results of our first two rounds.

The challenge allows participants to participate in the constrained MT or unconstrained MT. The latter allows for using additional training data and pre-trained models. In the first round, we only participated in the constrained translation task. We used a multilingual MT to train the English to French, Spanish, German Italian, and Swedish models. For comparison reasons, we also train bilingual models for English to German and French. As shown in Table 2, the English to French and Spanish models achieve significantly higher scores than the English to German, Italian and Swedish models. Since the number of training samples are not that much different, this could be due to the fact that the data for these two language pairs is the cleanest. As for the difference between bilingual and multilingual models, we can see that in the English to German direction, the multilingual model achieves a much higher score while on the English to French side, the difference is insignificant. Further experiments are needed to understand why this has happened, but previous studied have already shown that multilingual models doesn't bring much improvement to rich language pairs such as English and French.

2.4 Discussion

In the previous section, we described the participation of Lingua Custodia in the Covid-19 MLIA @ Eval Initiative. As our first attempt, we used a multilingual MT model and achieved promising results. In the rest of the challenge we plan to use different techniques such as oversampling low resourced languages, fine-tuning multilingual model on bilingual data.

3 Round 2

3.1 Data

In the second round, the organizers have added significantly more data to all language pairs particularly French, Spanish, German and Italian. In 3, statistics of round 2 data is given. In this round, the language with highest number of training samples is Spanish which has 2.8 million sentences and the language with the lowest number of training sentences in Arabic which has only 424K sentences. Based on the rules, both round 1 and 2 data can be used for the constrained task. This makes Spanish the language with the most training data,followed by French, German, Italian, Greek, Swedish and finally Arabic which has no data from round 1.

Table 3. Corpora statistics for round 2.

	German		French		Spanish		Italian		Modern Greek		Swedish		Arabic	
	En	De	En	Fr	En	Es	En	It	En	El	En	Sv	En	Ar
Train	1,536,411		2,412,653		2,862,002		1,026,064		673,961		374,998		424,434	
Validation	4,000		4,000		4,000		4,000		4,000		4,000		4,000	
Test	4,000		4,000		4,000		4,000		4,000		4,000		4,000	

3.2 Machine translation models

As for the second round, we follow a much more systematic process in order to find out best parameters that fit the task such as vocabulary size and several other pre-processing steps that we didn't include in the first round.

Pre-processing For the second, we follow a much more rigorous pre-pressessing scheme. We first tokenize the source and target sentences with Moses tokenizer. Then we filter-out samples where there is a big difference of length between source and target sentences and replace consecutive spaces with one single space. As for casing, instead of using true casing of Moses, we decided to try a new technique called inline-casing [2]. In this technique, every word is lower cased and in order to preserve the casing information, a tag indicating the original case of the word is placed after it. We use ¡U¿, ¡T¿ and ¡M¿ tags after upper cased, title cased and mixed cased words respectively. One difference between our implementation and [2] is that we apply casing before applying sub-word segmentation. In addition, for the sake of simplicity, we decided to apply this technique only to the source sentence. The final step of pre-processing is to append the language token to each source sentence in order to indicate the target language which is necessary in case of multilingual models.

Model architectures For the second round of the challenge we use Sockeye toolkit [4] as it has better support for training on multiple GPUs and handles data loading more efficiently than the Seq2SeqPy toolkit.

As in the first round we train both bilingual and multilingual models. However in this round we test several approaches in order to understand the extend of which multilingual MT model can benefit high as well as low resourced languages.

We first start by analyzing the best vocabulary size for the multilingual model. We test vocabulary sizes ranging from 16K up to 50K. Then, we test if pre-processing brings any improvement or not, this is because in the first round we applied minimal pre-processing and we were still able to achieve competitive results. Another reason of not applying pre-processing is because SentencePiece already applies tokenization and sub-word segmentation all at once.

Finally, we experiment with several ways of combining the data from multiple languages to train the multilingual model. The most straight approach is to combining all the data in a very naive way. The problem with this approach is that low resourced languages like Arabic will not be well represented enough and will be overshadowed by more resourceful languages. To overcome this issue, we propose to oversample the data of mid or less resourced languages (low resourced in the context of the challenge) like Italian, Swedish Greek and Arabic so that all languages have equal contributions to the model. Other than that, we also we also experiment with training similar languages together by excluding Arabic and Greek. The results are presented in the next section.

Hyper-parameters

As in the first round, we use the standard transformer architecture with 6 encoder and 6 decoder layers with the same size of the embedding, hidden states, feed-forward layer and attention heads. We use Adam optimizer with a learning rate of 0.0003 and a warm up step of 2000. All outputs are generated with a beam size of 5.

3.3 Experiments

The first set of experiments of the second round starts by testing different vocabulary sizes on a multilingual model covering all the 7 language pairs. We test vocabulary sizes of 16K, 20K, 30K, 40K and 50K. The results are presented in Table 3.3. As we can see, the blue score is lowest with the 16K vocab size and highest with 40K and 50K.

Table 4. Bleu scores on models with different vocabulary sizes on the concatenated dev set of all 7 languages.

Vocab size	Bleu score
16K	36.7
30K	38.4
40K	39.0
50K	39.0

The second experiment is about deciding if we need further pre-processing (described in 3.2) or Sentencepiece is enough by itself like in round 1. We train two multilingual models one where the data is pre-processed and the other one without pre-processing. Both models are trained with the vocabulary size of 40K.

Based on the results found in Table 3.3, we can see that pre-processing helps improving the results significantly. Thus, for rest of the experiments, we pre-process the data and set the vocabulary size to 40K.

Table 5. Bleu scores on models with and without pre-processing on the concatenated dev set of all 7 languages.

	Bleu score
w/o pre-processing	39.0
w/ pre-processing	43.2

Results on the dev set As in the first round, we only participated in the constrained task. However, in this round we perform three main types of models: bilingual, multilingual 7 language, and multilingual 5 language. The reason for having 5 and 7 multilingual models is because Arabic and Greek have different writing scripts than the other 5 languages and we wanted to remove those two in the 5 language model. In addition to this, for the multilingual model, we performed oversampling of the andmid abd low resourced languages. In the 7 language model, Italian, Swedish, Greek and Arabic have been oversampled, and in the 5 language one, only Italian and Swedish have been oversampled. Results presented in Table 3.3 shows some very interesting findings. Firstly, we can see that for the languages with most amount of data, i.e., French, German, and Spanish, bilingual models still achieve the highest bleu score. More interestingly, oversampling less resourced languages seems to always reduce the score of high resourced ones. The 5 language models seems to be only benefiting Italian and Swedish as they both achieve higher bleu score and both significantly go up with oversampling. Finally, when it comes to Greek and Arabic, they both benefit a lot from the multilinguality and again from oversampling.

Table 6. Bleu scores on the development set. ov indicates that less resourced languages were oversampled.

	En-De	En-Fr	En-Es	En-It	En-El	En-Sv	En-Ar
Bilingual	41.2	58.9	57.0	45.4	42.4	16.1	23.8
Multi-5lang	40.0	58.1	56.1	47.6	-	17.9	-
Multi-5lang-ov	38.3	57.0	55.3	48.1	-	21.9	-
Multi-7lang	38.9	57.3	55.5	47.4	44.5	17.7	25.5
Multi-7lang-ov	37.6	56.3	54.5	47.6	45.7	18.8	28.9

Results on the test set In this section we present the results of our submitted systems and their corresponding scores on the test set. Results are presented in Tables 7-13 for all the 7 language pairs. Translations are scored using Bleu score, Translation Error Rate (TER) and BEER [8] which is a learnt evaluation metric that is supposed to be highly correlated with human judgements. We perform small tweaks to some of our models before submitting. These tweaks include averaging best 4 checkpoints (called avg in the rest of the paper), fine-tuning the the multilingual model on bilingual data of a specific language pair (shown as ft in the tables), and as in the previous section ov means oversampling of mow resourced languages.

The results on the test set are very consistent with the ones on the development test. For high resourced languages like French and Spanish, bilingual models are always better than multilingual ones, except for German where we fine-tuned the 5lang model and managed to outperform the bilingual model. In Spanish and German, we achieve 1st position in the ranking and in French we achieve 3rd position. For Italian and Swedish, the oversampled and fine-tuned 5lang models obtain the best results. For Italian we rank 1st and 2nd in Swedish. Finally, for Arabic and Greek the oversampled 7lang models has the best bleu score. We rank 1st in Arabic and 2nd in Greek.

Table 7. English → German

System	Bleu	Ter	BEER
5lang-ft-avg	40.3	48.4	66.8
5lang-ft	39.8	48.9	66.5
1lang	39.7	50.1	65.9
7lang	38.6	50	65.8

Table 8. English → French

System	Bleu	Ter	BEER
1lang	57.2	34.9	74.5
7lang	55.8	35.7	73.9

Table 9. English → Spanish

System	Bleu	Ter	BEER
1lang-avg	56.6	33.7	75.2
5lang-ft-avg	56	33.8	75.1
7lang	55.3	34.4	74.8

Table 10. English → Italian

System	Bleu	Ter	BEER
5lang-ov-ft-avg	48.9	40.3	70.2
5lang-ov	48	40.9	69.8
1lang	45.3	44.1	67.8

Table 11. English → Greek

System	Bleu	Ter	BEER
7lang-ov-ft-avg	44.7	43.8	67.2
7lang-ov	44.2	44.1	67
7lang	43.2	44.8	66.5
1lang	41.2	47.3	64.8

Table 12. English → Swedish

System	Bleu	Ter	BEER
5lang-ov-ft-avg	22	71.7	49.2
5lang-ov	21.8	71.5	49.4
7lang-ov	18.3	74.9	47.4
5lang	17.7	75.6	47
1lang	16.7	78.9	45.4

Table 13. English → Arabic

System	Bleu	Ter	BEER
7lang-ov	25.1	64.7	57.6
7lang	22	67.4	55.8
1lang	19.1	73.8	53

3.4 Discussion

In the second round of the Covid-19 MLIA Evaluation Initiative, we participated in all the 7 language pairs of the constrained task. Our main goal was to test how oversampling of less resourced languages behave. Based on the results that we have achieved, we can say that oversampling is very important for languages where there is a limited data. In addition, we also wanted to test different multilingual models with different number of languages. As we saw from the results, some it is better to remove certain languages in order to boost the performance of other languages like removing Greek and Arabic benefited Swedish and Italian in our case.

In the next round we are planning to test adapter modules [1].

References

[1] Bapna, A., Arivazhagan, N., Firat, O.: Simple, scalable adaptation for neural machine translation. arXiv preprint arXiv:1909.08478 (2019)

[2] Berard, A., Calapodescu, I., Roux, C.: Naver labs europe's systems for the wmt19 machine translation robustness task. arXiv preprint arXiv:1907.06488 (2019)

[3] Casacuberta, F., Ceausu, A., Choukri, K., Deligiannis, M., Domingo, M., García-Martínez, M., Herranz, M., Papavassiliou, V., Piperidis, S., Prokopidis, P., Roussis, D.: The Covid-19 MLIA @ Eval initiative: Overview of the machine translation task. https://bitbucket.org/covid19-mlia/organizers-task3/src/master/report/ (2021)

[4] Domhan, T., Denkowski, M., Vilar, D., Niu, X., Hieber, F., Heafield, K.: The sockeye 2 neural machine translation toolkit at amta 2020. arXiv preprint arXiv:2008.04885 (2020)

[5] Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al.: Moses: Open source toolkit for statistical machine translation. In: Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions, pp. 177–180, Association for Computational Linguistics (2007)

[6] Kudo, T., Richardson, J.: Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. arXiv preprint arXiv:1808.06226 (2018)

[7] Qader, R., Portet, F., Labbé, C.: Seq2seqpy: A lightweight and customizable toolkit for neural sequence-to-sequence modeling. In: LREC 2020, pp. 7140–7144 (2020)

[8] Stanojević, M., Sima'an, K.: Evaluating mt systems with beer. The Prague Bulletin of Mathematical Linguistics **104**, 17–26 (2015)

[9] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30**, 5998–6008 (2017)