CANINE: Pre-training an Efficient Tokenization-Free Encoder for Language Representation
Machine Learning for Natural Language Processing 2020

Chloé SEKKAT
ENSAE & ENS Paris-Saclay
chloe.sekkat@ensae.fr

Jocelyn BEAUMANOIR
ENSAE & ESSEC
jocelyn.beaumanoir@ensae.fr

1 Problem Framing

In this project we chose to take a **scientific** approach in which we want to evaluate the performances of CANINE (Clark et al., 2021), the **first pre-trained tokenization-free and vocabulary-free encoder**, that operates directly on character sequences without explicit tokenization. It seeks to generalize beyond the orthographic forms encountered during pre-training. We want to compare its performances on several downstream tasks and against several SOTA models. We will focus on two main fields: Question Answering (QA) & Sentiment Analysis/Classification (SC).

For QA, we are interested in 5 downstream tasks: extractive QA on SQuADv2, generalization in zero-shot transfer settings on multilingual data, robustness to noise, domain adaptation and resistance to adversarial attacks.

For SC, we are also interested in various downstream tasks: binary classification with SST-2 dataset, robustness to noise, binary classification using more natural real-life noise (Sentiment140) and zero-shot transfer on multilingual data (MARC (Keung et al., 2020)).

Due to limited space, we shall **not** described all our experiments in this report. If you are interested, we strongly encourage you to checkout **our GitHub** [1].

2 Experiments Protocol

The general method employed can be summarized as follows:

- set seed (for reproducibility)
- encode the dataset using the tokenizer associated to each model (proposed by HuggingFace)
- fed the tokenized data to the model
- training/evaluation loop
- monitor validation loss/accuracy
- use early stopping
- predict/evaluation on test set once the best model has been found
- analyse the predictions and errors of the model
- compare to other models & build intuition

CANINE-S (subword loss) and CANINE-C (Autoregressive Character Loss) will be compared to: BERT (Devlin et al., 2019), mBERT, RoBERTa (Liu et al., 2019), XLM-R (Conneau et al., 2019) and DistilBERT (Sanh et al., 2019).

For QA tasks, we added a regression head to each model. It is one linear layer (of shape (768,2) for CANINE for instance) to predict start and end logits for minimal span answer (given a context and a question, extract the passage with the answer in the context). Note that the trickiest part was to implement a functional tokenizer for CANINE since the one provided by HF is not suited for QA.

For SC tasks, we added a classification head to each model. It is a linear layer (of shape (768, *num_labels*) for CANINE e.g.) to predict logits over the number of classes in the dataset (2 for SST2 and 3 for Sentiment140), with a dropout layer with $p = 0.1$ or 0.2 depending on the model. The implementation was quite straightforward and in the final pipeline we used the architecture provided by HF in order to keep only pipeline clean and fast.

3 Exploratory Data Analysis

For this project, we used various datasets. For some of them, we did an exploratory analysis which can be found here. You can also find the main figures in Appendix A.

4 Results

As the goal is to evaluate CANINE and compare its performances to BERT-like models, these mod-

[1] https://github.com/chloeskt/nlp_ensae

els are our baseline. For QA tasks, we focused on F1 score and Exact Match while for SC we looked at the accuracy of each model.

4.1 Question Answering

We experimented on 5 different sub-tasks. When finetuning on SQuADv2 (Rajpurkar et al., 2016), CANINE performs decently well but other models are often better (Table 5). When evaluated in zero-shot transfer setting on XQuAD (Artetxe et al., 2019), CANINE does not perform well (Tables 6, 7). The gap between XLM-RoBERTa and CANINE-C increases when evaluated on languages such as Vietnamese, Thai or Chinese. These languages are mostly isolating ones i.e. language with a morpheme per word ratio close to one and almost no inflectional morphology. In the few-shot learning and domain adaptation experiment with CUAD (Hendrycks et al., 2021), CANINE performs similarly as the other models. Note that here, multilingual models are better than unilingual ones even if CUAD is in English. Finally, CANINE is not robust to adversarial attacks in the few-shot settings (dynabench/qa dataset (Bartolo et al., 2020), Table 10 - no time to train on full dataset).

4.2 Sentiment Classification

Again we experimented on 5 sub-tasks. When finetuned on SST2 (Socher et al., 2013) for binary classification, CANINE is comparable to BERT-like models on the validation set but not on the test set (Table 13). When artificial noise is added only to val and test sets, CANINE does not outperform other models (Table 14), it is not robust enough. When trained on a more challenging dataset, Sentiment140, CANINE is similar to mBERT, but here unilingual models outperforms multilingual ones (Table 15). When we only evaluate models, finetuned on SST2, on Sentiment140, performances decrease (as expected), but it decrease more for CANINE and mBERT than for other models (Table 16). In zero-shot transfer on multilingual data, we consider MARC data for French, German, Japanese and Chinese (chosen for their linguistic properties). CANINE-S is similar to mBERT for French and Chinese data (Table 17) but worse for German and Japanese. XLM-RoBERTa is extremely better than other models in this setting (+10pp). When finetuned on the corresponding training data (Table 18), CANINE-S is similar to XLM-R and mBERT but for languages with a smaller genetic proximity to English (Table 2), XLM-R is better.

5 Discussion/Conclusion

In our zero-shot transfer QA experiments, CANINE does not appear to perform as well as token-based transformers such as mBERT. It might be because it was finetuned on English (analytical language) and hence cannot adapt well in zero-shot transfer especially to isolating languages (Thai, Chinese) and synthetic ones with agglutinative morphology (Turkish) or non-concatenative (Arabic). CANINE works decently well for languages close enough to English, e.g. Spanish or German. While mBERT and CANINE have both been pretrained on the top 104 languages with the largest Wikipedia using a MLM objective, XLM-R was pretrained on 2.5TB of filtered CommonCrawl data containing 100 languages. This might be a confounding variable. Also, CANINE-S seems to be robust to high level of artificial noise and even slightly better than BERT and mBERT. Finally, one might also note that multilingual model do, overall, have better capacities of generalization and better scores on these QA tasks hinting that there are meaningful information in cross-lingual representations.

From our SC experiments, overall, other BERT-like models were better than CANINE. However, note that on most tasks, CANINE performs similarly to mBERT and might be even slightly better. But for general tasks in English or more generally with languages for which we have a lot of resources, XLM-R and/or RoBERTa are better. We were not able to prove in our experiments that CANINE is better than tokenizer-based BERT-like models even on more challenging and complex languages such as Thai, Chinese, Japanese or Arabic. When finetuning for binary classification on German, Japanese and Chinese, CANINE-S was slightly better than XLM-R on German (high proximity with English, West Germanic family). But this was not the case in Japanese and Chinese where mBERT and XLM-R should be prefered (+3pp in accuracy).

To go further, we could try to evaluate the importance of the chosen tokenizer for each BERT-like model and how it impacts their final predictions. This would allow us to quantify what is captured and what is missing by each model at this level and compare that to the character level embeddings of CANINE.

References

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *arXiv e-prints*, page arXiv:1606.05250.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2019. On the cross-lingual transferability of monolingual representations. *CoRR*, abs/1910.11856.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

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

Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. 2020. Beat the AI: Investigating adversarial human annotation for reading comprehension. *Transactions of the Association for Computational Linguistics*, 8:662–678.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. 2020. The multilingual amazon reviews corpus.

Jonathan H. Clark, Dan Garrette, Iulia Turc, and John Wieting. 2021. CANINE: Pre-training an efficient tokenization-free encoder for language representation.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. Cuad: An expert-annotated nlp dataset for legal contract review.

Appendix

A Exploratory Data Analysis

A.1 Illustration of Zipf Law

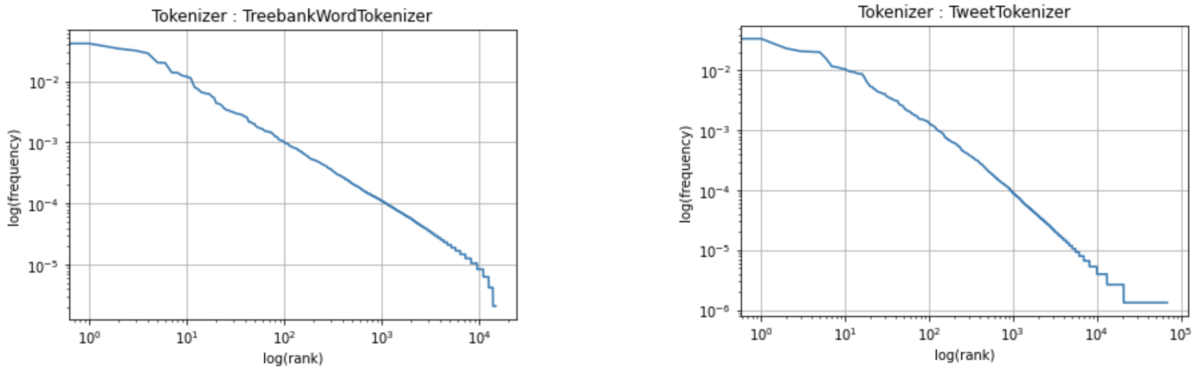

Figure 1: Illustration of Zipf Law on SST2 (right) with TreeBankTokenizer and Sentiment 140 (left) with TweetTokenizer

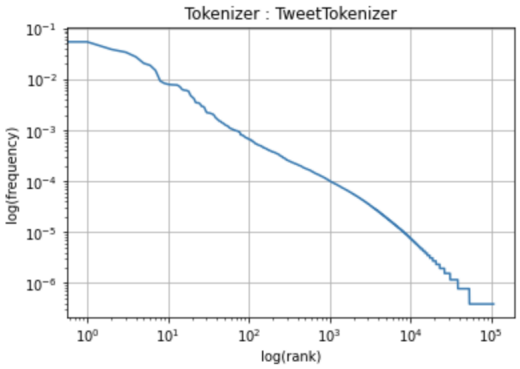

Figure 2: Zipf Law respected on SQuADv2 dataset, using TweetTokenizer

A.2 Sentiment Labels

	Negative	**Positive**
SST2	29780	37569
Sentiment140	800000	800000

Table 1: Number of positive and negative sentences in the original SST2 and Sentiment140 training sets.

B Language family

Language	Family	Proximity with English
Russian	East Slavic	60.3
Dutch	West Germanic	27.2
German	West Germanic	30.8
Turkish	Turkic	92.0
French	Romance	48.7
Thai	Tai	92.9
Chinese	Sino-Tibetan	82.4
Japanese	Japanese	88.3
Romanian	Romance	54.9
Greek	Greek	69.9
Spanish	Romance	57.0
Hindi	Indic	65.2
Arabic	Semitic	83.6

Table 2: Language family and genetic proximity between languages and English.

C Question Answering

C.1 Datasets

The datasets splits are as follows:

Nb of samples	Training	Validation	Test
SQuADv2	130 319	10 686	1 187
SQuADv1.1	87 599	10 570	-
XQuAD (per language)	-	1 190	-
CUAD	224	-	656
dynabench/qa	200	-	600

Table 3: Number of samples per splits and datasets.

C.2 Training hyperparameters

	RoBERTa	BERT	DistilBERT	mBERT	XLM-ROBERTA	CANINE-c	CANINE-s
Batch size	12	8	8	8	8	4	4
Learning Rate	2e-5	3e-5	3e-5	2e-5	3e-5	5e-5	5e-5
Weigh decay	1e-4	0	1e-2	0	0	0.01	0.001
Nb of epochs	3	2	2	2	2	3	2.5
Number of training examples	131823	131754	131754	132335	133317	130303	130303
Number of validation examples	12165	12134	12134	12245	12360	11861	11861
Max sequence length	348	348	348	348	348	2048	2048
Doc stride	128	128	128	128	128	512	512
Max answer length	30	30	30	30	30	256	256
Lr scheduler	cosine	none	linear	none	none	linear	linear
Warmup ratio	0.1	0	0.1	0	0	0.1	0.1

Table 4: Extensive list of hyperparameters used when finetuning models on SQuADv2 dataset. Note that models were trained on one Tesla P100 (16GB).

C.3 Extractive QA on SQuADv2

In this settings, CANINE performs decently well (especially CANINE-c i.e. CANINE trained with Autoregressive Character Loss).

	F1-score	EM score
BERT	76.74	73.59
RoBERTa	**82.02**	**78.54**
DistilBERT	67.81	64.71
CANINE-C	74.1	69.2
CANINE-S	72.5	69.6
mBERT	77.51	74.1
XLM-RoBERTa	78.3	75.12

Table 5: F1 and Exact Match scores obtained on SQuADv2 manually created test sets. Results are averaged over three runs.

C.4 Zero-shot transfer on multilingual dataset

In this setting, CANINE does not perform very well. On average it is -20 F1 lower than XLM-RoBERTa and -10 F1 lower than mBERT even if we expected CANINE to perfom better since it operates on characters and hence is free of the constraints of manually engineered tokenizers (which often do not work well for some languages e.g. for languages that do not use whitespaces such as Thai or Chinese) and fixed vocabulary. The gap between XLM-RoBERTa and CANINE-C increases when evaluated on languages such as Vietnamese, Thai or Chinese. These languages are mostly isolating ones i.e. language with a morpheme per word ratio close to one and almost no inflectional morphology.

	CANINE-C	CANINE-S	mBERT-base	BERT-base	XLM-RoBERTa
English	78,77	79,03	83,59	82,3	**82,8**
Arabic	43,78	29,74	54,09	11,76	**62,48**
German	59,57	55,35	68,4	19,41	**72,47**
Greek	46,93	30,82	56,47	10,21	**70,93**
Spanish	60,47	59,48	72,84	19,72	**75,18**
Hindi	35,21	30,93	51,06	11,07	**62,1**
Russian	60,49	55,09	68,33	9,47	**73,12**
Thai	**37,28**	31,2	**27,63**	10,04	**65,21**
Turkish	31,09	23,83	44,62	16,76	**65,34**
Vietmanese	43,14	35,52	64,49	24,63	**73,44**
Chinese	34,86	28,68	52,71	8,15	**65,68**
Romanian	56,62	43,69	69,31	20,03	**74,78**
Average	**49,02**	41,95	59,46	20,30	**69,16**

Table 6: F1-scores on XQUAD dataset in zero-shot transfer setting. Results are averaged over 3 runs.

	CANINE-C	CANINE-S	mBERT-base	BERT-base	XLM-RoBERTa
English	67,38	66,34	79,51	69,57	**72,18**
Arabic	26,25	13,75	37,22	4	**45,79**
German	43,16	38,27	50,84	4,9	**55,21**
Greek	29,14	13,42	40,16	5,37	**53,19**
Spanish	42,74	39,57	54,45	4,7	**56,3**
Hindi	18,93	16,54	36,97	4,8	**45,042**
Russian	43,48	35,65	52,1	4,62	**55,54**
Thai	20,5	17,91	21,26	2,6	**54,28**
Turkish	14,8	10,11	29,41	4,87	**48,85**
Vietmanese	25,17	19,65	45,21	7,64	**54,02**
Chinese	21,36	20,2	42,26	3,1	**55,63**
Romanian	39,98	26,5	54,62	6,21	**61,26**
Average	**32,74**	26,49	45,33	10,20	**53,19**

Table 7: Exact-Match scores on XQUAD dataset in zero-shot transfer setting. Results are averaged over 3 runs.

C.5 Robustness to noise

In this experience, the goal is to evaluate the models' robustness of noise. To do so, we created 3 noisy versions of the SQuADv2 dataset where the questions have been artificially enhanced with noisy (in our

case we chose RANDOMCHARAUG from NLPAUG library with action SUBSTITUTE but in our package 4 other types of noise have been developed).

Three levels of noise were chosen: 10%, 20% and 40% . Each word gets transformed with probability p into a misspelled version of it (see NLPAUG documentation [2] for more information).

The noise is **only** applied to the test set (on SQuADv2) made of 1187 examples. We compared the 7 models we finetuned on the clean version of SQuADv2 (first experiment) on these 3 noisy datasets (on for each level of p). The following table gathers the results (averaged over 3 runs):

	Noise level 10%		**Noise level 20%**		**Noise level 40%**	
	F1 score	**EM**	**F1 score**	**EM**	**F1 score**	**EM**
BERT	73,68	70,79	71,22	68,55	66,42	63,74
RoBERTa	**79,06**	75,87	**76,57**	73,56	**70,7**	68,18
DistilBERT	65,85	63,05	64,42	61,92	60,77	58,78
mBERT	74	70,75	71,66	68,46	67,08	64,74
XLM-RoBERTa	74,54	71,61	**72,68**	69,81	67,12	64,43
CANINE-C	69,64	66,89	67,88	65,43	66,03	63,9
CANINE-S	72,25	69,65	70,3	68,03	**67,18**	64,6

Table 8: F1 and EM scores reported on three noisy test sets extracted from SQuADv2. RoBERTa is the most robust model. For high level of noise, CANINE-S performs relatively well. Results are averaged over three runs.

C.6 Few-shot learning and domain adaptation

The goal of this experiment is to measure the ability of CANINE (and other models) to transfer to unseen data, in another domain. This could either be done in zero-shot or few-shot settings. Here we decided to go with the latter as it is more realistic. In real life, a company might already have a custom small database of labeled documents and questions associated (manually created) but would want to deploy a Question Answering system on the whole unlabeled database. The CUAD dataset is perfect for this task as it is highly specialized (legal domain, legal contract review). The training set is made of 22450 question/context pairs and the test set of 4182. We randomly selected 1% of the training set (224 examples) to train on for 3 epochs, using the previously finetuned models on SQuADv2. Then each model was evaluated on 656 test examples. Results are reported in the following table and to ensure fair comparison, all models where trained and tested on the exact same examples.

	F1 score	**EM score**
BERT	74.18	72.72
RoBERTa	73.83	72.24
DistilBERT	72.86	71.37
mBERT	74.50	73.12
XLM-RoBERTa	**76.64**	73.44
CANINE-C	**72.51**	71.39
CANINE-S	**72.27**	71.27

Table 9: F1 scores and EM scores on 656 examples of CUAD's test set. Models have been trained for 3 epochs on 224 training examples. Results are averaged over three runs.

CANINE-S and CANINE-C performances are close to those of DistilBERT and above those of BERT.

C.7 Few-shot learning and adversarial attacks

This last Question Answering-related experiment aims at testing CANINE abilities not to be fooled in adversarial settings. We decided to us the dynabench/QA dataset (BERT-version). The latter is an adversarially collected Reading Comprehension dataset spanning over multiple rounds of data collect. It has been made so that SOTA NLP models find it challenging.

[2] https://github.com/makcedward/nlpaug/blob/master/nlpaug/augmenter/char/random.py

We decided to take models finetuned on SQuADv2, take 200 examples (2%) extracted from dynabench/qa training set to train each model for 3 epochs and then evaluate these models on 600 test examples (60% of the full test set).Our results are displayed in the following table. Again, to ensure fair comparison, all models are trained on the exact same examples and evaluated on the same ones.

	F1 score	EM score
BERT	38.13	25.6
RoBERTa	**47.47**	35.8
DistilBERT	32.64	22.5
mBERT	38.43	28.6
XLM-RoBERTa	36.51	27.6
CANINE-C	**28.25**	18.6
CANINE-S	**27.40**	17.2

Table 10: F1 scores and EM scores on 600 test examples from dynabench/qa dataset. Models have been trained for 3 epochs on 200 training examples. Results are averaged over three runs.

Both CANINE-S and CANINE-c perform very badly and are not able to resist such attacks.

D Sentiment Classification

D.1 Datasets

	Train	Validation	Test
SST2	63981	3368	872
Sentiment140	63360	16000	359
Amazon Reviews Multilingual (per language)	160000	4000	4000

Table 11: Number of samples per split and per datasets.

D.2 Training hyperparameters

	RoBERTa	BERT	DistilBERT	mBERT	XLM-ROBERTA	CANINE-c	CANINE-s
Batch size	12	12	12	12	12	6	6
Learning Rate	2e-5	2e-5	2e-5	2e-5	2e-5	2e-5	2e-5
Weigh decay	1e-2	1e-2	1e-2	1e-2	1e-2	1e-2	1e-2
Nb of epochs	2	2	2	2	2	2	2
Number of training examples	63981	63981	63981	63981	63981	63981	63981
Number of validation examples	872	872	872	872	872	872	872
Lr scheduler	linear	linear	linear	linear	linear	linear	linear
Warmup ratio	0.1	0.1	0.1	0.1	0.1	0.1	0.1

Table 12: Extensive list of hyperparameters used when finetuning models on SST2 dataset. Note that models were trained on one Tesla P100 (16GB).

D.3 Sentiment classification on benchmark SST2 dataset

D.4 Robustness to noise

In this experience, the goal is to evaluate the models' robustness of noise. To do so, we created 3 noisy versions of the SST2 dataset where the sentences have been artificially enhanced with noisy (in our case we chose RANDOMCHARAUG from NLPAUG library with action 'substitute' but in our package 4 other types of noise have been developed).

Three levels of noise were chosen: 10%, 20% and 40% . Each word gets transformed with probability p into a misspelled version of it.

The noise is **only** applied to the SST2 validation and test sets made of 3368 and 872 examples respectively. We compared the 7 models we finetuned on the clean version of SST2 (first experiment) on these 3 noisy datasets (on for each level of p). The following table gathers the results (averaged over 3 runs):

	Val set	Test set
BERT	0.94	0.93
RoBERTa	0.94	0.94
DistilBERT	0.94	0.91
mBERT	0.93	0.88
XLM-RoBERTa	0.94	0.90
CANINE-C	0.93	0.86
CANINE-S	0.92	0.85

Table 13: Accuracy on validation and test sets of SST2 dataset. Results are averaged over three runs. Both CANINE models perform decently well on the validation set (3368 examples) but have more difficulties on the test set (872 examples). mBERT has similar behavior.

	Noise level 10%		Noise level 20%		Noise level 40%	
	Val set	Test set	Val set	Test set	Val set	Test set
BERT	0.88	0.87	0.85	0.82	0.80	0.80
RoBERTa	0.88	**0.89**	**0.87**	**0.85**	**0.83**	**0.82**
DistilBERT	0.85	0.82	0.82	0.79	0.76	0.76
mBERT	0.88	0.82	0.85	0.80	0.80	0.76
XLM-RoBERTa	**0.89**	0.85	0.86	0.83	0.81	0.81
CANINE-C	0.86	0.80	0.83	0.76	0.79	0.74
CANINE-S	0.85	0.80	0.83	0.77	0.78	0.74

Table 14: Accuracy reported on three noisy validation and test sets extracted from SST2 (models were finetuned on clean train and validation sets). RoBERTa is the most robust model. For high level of noise, CANINE perform relatively well compared to DistilBERT. Results are averaged over three runs.

Both CANINE models have a better performance than DistilBERT for a high level of noise (¿= 40%). However all other models are better to handle this type of artificial noise, RoBERTa being the best of all.

D.5 Sentiment Classification on more challenging Sentiment140 dataset (tweets)

The following experience is meant to evaluate the performances of the various models on a more challenging dataset: Sentiment140. This dataset is made of 1.6 million of tweets, all in English. The language used is very different from the one in SST2 as it is made of more abbreviations, colloquialisms, slang, etc. Therefore it is expected to be hard for the models to handle such text (which is "naturally" noisy). CANINE has a theoretical advantage on such dataset due to the fact that it is tokenizer-free and operates at the character level.

The following table reports the results we obtained when finetuning all models on the (smaller) training set of 63360 examples.

	Val set	Test set
BERT	0.84	0.86
RoBERTa	0.87	0.86
DistilBERT	0.83	0.85
mBERT	0.79	0.78
XLM-RoBERTa	0.81	0.80
CANINE-C	0.79	0.78
CANINE-S	0.80	0.79

Table 15: Accuracy on validation and test sets of Sentiment140 with **models finetuned on Sentiment140 training set**. Both CANINE models perform similarly to mBERT. Note that multilingual models are worse than unilingual dataset on this English dataset. This might be explained by the fact that there is natural noise in the dataset (tweets). Results are averaged over three runs.

D.6 Zero-shot transfer learning and domain adaptation from SST2 to Sentiment140

In this experience we would like to see how CANINE models perform when they are faced with "natural" noise that they were **not** trained on. Compared to the previous experience where models where trained on Sentiment140, here models are trained on SST2 but evaluated on validation and test set from Sentiment140.

In the previous task, CANINE models were not the best performing one. Actually, with mBERT, they were the last ones. Here we are evaluating something different: the ability for a model to adapt to another domain (in the sense that the topic and the way of writing/speaking are different) in a zero-shot transfer setting. It might be that, in real life settings, one has access to a clean benchmark-type dataset (such as SST2) but wants to do inference on a dataset whose subject is quite different and full of misspellings and grammar errors.

Results are reported in the following table:

	Val set	**Test set**
BERT	0.72	0.84
RoBERTa	0.73	0.88
DistilBERT	0.71	0.82
mBERT	0.68	0.76
XLM-RoBERTa	0.72	0.83
CANINE-C	0.64	0.77
CANINE-S	0.64	0.73

Table 16: Accuracy on validation and test sets of Sentiment140, with **models finetuned on SST2 only**. CANINE models do not perform well on this domain adaptation in zero-shot learning setting. Results are average over three runs.

CANINE models do not perform well on this task. They have -9 percentage point of accuracy compared to RoBERTa for instance (best performing model on this task) on the validation set. We noticed that mBERT has more difficulties than other BERT-like models on Sentiment140 dataset overall. Again, CANINE and mBERT have similar behavior.

D.7 Zero-shot transfer learning on multi-lingual data

This experiment builds on the idea that CANINE is expected to perform better on languages with a different morphology than English, for instance on non-concatenative morphology (such as Arabic and Hebrew), compounding (such as German and Japanese), vowel harmony (Finnish), etc. Moreover, it is known that splitting on whitespaces (which is often done in most tokenizer - note that SentencePiece has an option to skip whitespace splitting) is not adapted to languages such as Thai or Chinese.

In this experience, models have been finetuned on the English dataset SST2 and are only evaluated both on validation and tests sets of 4 languages from the Multilingual Amazon Reviews Corpus (MARC, (Keung et al., 2020)). We considered the four following language: German, French, Japanese and Chinese for their morphological properties.

This dataset contains for each review the number of stars associated by the reviewer. To derive positive/negative sentiment from this, we considered that if 1 or 2 stars only have been associated to the review, the sentiment is negative. While if 4 or 5 stars have been chosen, the review is positive. Neutral reviews, with 3 stars, were not considered. For each language, this gives us 160000 training samples, 4000 validation samples and 4000 test samples.

CANINE-S is similar to mBERT for French and Chinese data. Overall XLM-RoBERTa is extremely better than other models. Note that its pre-training strategy is different from the one of mBERT and CANINE. Indeed, while mBERT and CANINE have both been pretrained on the top 104 languages with the largest Wikipedia using a MLM objective, XLM-RoBERTa was pretrained on 2.5TB of filtered CommonCrawl data containing 100 languages. This might be a confounding variable.

	French		German		Japanese		Chinese	
	Val set	**Test set**	**Val set**	**Test set**	**Val set**	**Test set**	**Val set**	**Test set**
mBERT	0.71	0.70	0.66	0.66	0.56	0.55	0.58	0.59
XLM-RoBERTa	0.87	0.86	0.86	0.87	0.87	0.85	0.80	0.79
CANINE-C	0.70	0.69	0.59	0.58	0.50	0.50	0.57	0.55
CANINE-S	0.71	0.70	0.61	0.61	0.52	0.52	0.57	0.57

Table 17: Accuracy on validation and test sets of MARC dataset, **models have been finetuned only on English SST2**. Results are averaged over three runs.

D.8 Finetuning on multilingual data

In this last experiment, we now compare CANINE to other BERT-like models on multilingual data where they are finetuned on it. This is the difference with the previous experience. To do so, we have chosen to work again with the MARC dataset, using data in German, Japanese and Chinese. We would like to see how CANINE compares and if it is better on languages which are more challenging for token-based models (Chinese for instance).

Please note that due to time and compute constraints, we considered only one CANINE model, CANINE-S.

	German		Japanese		Chinese	
	Val set	**Test set**	**Val set**	**Test set**	**Val set**	**Test set**
mBERT	0.93	0.93	0.92	0.92	0.87	0.88
XLM-RoBERTa	0.92	0.92	0.93	0.93	0.88	0.88
CANINE-S	0.93	0.93	0.90	0.89	0.85	0.85

Table 18: Accuracy on validation and test sets of MARC dataset, **models have been finetuned on the respective MARC training set for each language**. Results are averaged over three runs.

Quite surprisingly, on German, CANINE-S is slightly better than XLM-RoBERTa and has similar performance than mBERT. However on Japanese and Chinese, it is not the case. mBERT and especially XLM-RoBERTa should be preferred has they provide better accuracy on both validation and test sets.