Can You Traducir This? Machine Translation for Code-Switched Input

Jitao Xu, François Yvon

Can You Traducir This? Machine Translation for Code-Switched Input

Jitao Xu
Univ. Paris-Saclay,
& CNRS, LISN
Orsay, France
jitao.xu@limsi.fr

François Yvon
Univ. Paris-Saclay,
& CNRS, LISN
Orsay, France
francois.yvon@limsi.fr

Abstract

Code-Switching (CSW) is a common phenomenon that occurs in multilingual geographic or social contexts, which raises challenging problems for natural language processing tools. We focus here on Machine Translation (MT) of CSW texts, where we aim to simultaneously disentangle and translate the two mixed languages. Due to the lack of actual translated CSW data, we generate artificial training data from regular parallel texts. Experiments show this training strategy yields MT systems that surpass multilingual systems for code-switched texts. These results are confirmed in an alternative task aimed at providing contextual translations for a L2 writing assistant.

1 Introduction

Code-Switching (CSW) denotes the alternation of two languages within a single utterance (Poplack, 1980; Sitaram et al., 2019). It is a common communicative phenomenon that occurs in multilingual communities during spoken and written interactions. CSW is a well studied phenomenon in linguistic circles and has given rise to a number of theories regarding the structure of mixed language fragments (Poplack, 1978; Pfaff, 1979; Poplack, 1980; Belazi et al., 1994; Myers-Scotton, 1997). The Matrix Language Frame (MLF) theory (Myers-Scotton, 1997) defines the concept of *matrix* and *embedded* languages where the *matrix language* is the main language that the sentence structure should conform to and notably provides the syntactic morphemes, while the influence of the *embedded language* is lesser and is mostly manifested in the insertion of content morphemes.

The rise of social media and user-generated content has made written instances of code-switched language more visible. It is estimated that as much as 17% of Indian Facebook posts (Bali et al., 2014) and 3.5% of all tweets (Rijhwani et al., 2017) are code-switched. This phenomenon is also becoming more pervasive in short text messages, chats, blogs, and the like (Samih et al., 2016). Code-switching however remains understudied in natural language processing (NLP) (Aguilar and Solorio, 2020), and most work to date has focused on token-level language identification (LID) (Samih et al., 2016) and on language models for Automatic Speech Recognition (Winata et al., 2019). More tasks are being considered lately, such as Named Entity Recognition (Aguilar et al., 2018), Part-of-Speech tagging (Ball and Garrette, 2018) or Sentiment Analysis (Patwa et al., 2020).

We focus here on another task for CSW texts: Machine Translation (MT). The advent of Neural Machine Translation (NMT) technologies (Bahdanau et al., 2015; Vaswani et al., 2017) has made it possible to design multilingual models capable of translating from multiple source languages into multiple target languages (Firat et al., 2016; Johnson et al., 2017), where however both the input and output are monolingual. We study here the ability of such architectures to translate fragments freely mixing a "matrix" and an "embedded" language into monolingual utterances.

Our main contribution is to show that for the two pairs of languages considered (French-English and Spanish-English): (a) translation of CSW texts is almost as good as the translation of monolingual texts – a performance that bilingual systems are unable to match; (b) such results can be obtained by training solely with artificial data; (c) CSW translation systems achieve a near deterministic ability to recopy in the output target words found in the input, suggesting that they are endowed with some language identification abilities. Using these models, we are also able to obtain competitive results on the SemEval 2014 Task 5: L2 Writing Assistant, which we see as one potential application area of CSW translation.

2 Building translation systems for code-switched data

2.1 Code-switched data generation

Parallel corpora with natural CSW data are very scarce (Menacer et al., 2019) and, similar to Song et al. (2019a), we generate artificial CSW parallel sentences from regular translation data.

We first compute word alignments between parallel sentences using `fast_align`[1] (Dyer et al., 2013). We then extract so-called *minimal alignment units* following the approach of Crego et al. (2005): these correspond to small bilingual phrase pairs (e, f) extracted from (symmetrized) word alignments such that all alignment links outgoing from words in e reach a word in f, and vice-versa.

For each pair of parallel sentence, we first randomly select the matrix language;[2] then the number of replacements r to appear in a derived CSW sentence with an exponential distribution as:

$$P(r = k) = \frac{1}{2^{k+1}} \quad \forall k = 1, \ldots, \text{rep} \qquad (1)$$

where rep is a predefined maximum number of replacements. We also make sure that the number of replacements does not exceed half of either the original source or target sentences length, adjusting the actual number of replacements as:

$$n = \min(\frac{S}{2}, \frac{T}{2}, r) \qquad (2)$$

where S and T are respectively the length of the source and target sentences. We finally choose uniformly at random r alignment units and replace these fragments in the matrix language by their counterpart in the embedded language. Figure 1 displays examples of generated CSW sentences.

2.2 Machine translation for CSW data

2.2.1 Data preparation

We use WMT data for CSW data generation and for training MT systems. We discard sentences which do not possess the correct language by using the `fasttext` LID model[3] (Bojanowski et al., 2017). We use Moses tools (Koehn et al., 2007) to normalize punctuations, remove non-printing characters and discard sentence pairs with a source / target ratio higher than 1.5, with a maximum sentence length of 250. We tokenize all WMT data using Moses tokenizer.[4] Our procedure for artificial CSW data generation uses WMT13 En-Es parallel data with 14.5M sentences. For En-Fr, we use all WMT14 parallel data, for a grand total of 33.9M sentences. Our development sets are respectively `newstest2011` and `newstest2012` for En-Es, and `newstest2012` and `newstest2013` as development sets for En-Fr; the corresponding test sets are `newstest2013` (En-Es) and `newstest2014` (En-Fr).

2.2.2 Machine Translation systems

We use the `fairseq`[5] (Ott et al., 2019) implementation of Transformer base (Vaswani et al., 2017) for our models with a hidden size of 512 and a feed-forward size of 2048. We optimize with Adam, set up with an initial learning rate of 0.0007 and an inverse square root weight decay schedule, as well as 4000 warmup steps. All models were trained with mixed precision and a batch size of 8192 tokens for 300k iterations on 4 V100 GPUs. For each language pair, we use a shared source-target inventory built with Byte Pair Encoding (BPE) of 32K merge operations, using the implementation published by Sennrich et al. (2016).[6] Note that we do not share the embedding matrices. Our experiments with sharing the decoder's input and output embeddings or sharing all encoder+decoder embeddings did not yield further gains.

We compare three settings for `Code-Switch` models:

- the `base-csw` setting, where we train two separate systems, one translating CSW into English, and the other translating CSW into Spanish or French.
- the `multi-csw` setting, where we train one model able to generate either pure matrix or embedded language in the output. To this

[1] https://github.com/clab/fast_align

[2] Note that we abuse here the terms "matrix" and "embedded" language, as we do not attempt to generate linguistically realistic CSW data matching the constraints of the MLF theory. We use these terms in a much looser sense where the sentence in the "matrix" language is the one that receives arbitrary insertions from the "embedded" language. This means that our artificial CSW sentences will contain insertions of unconstrained fragments containing both content and function words, which the theory would generally consider ungrammatical.

[3] https://dl.fbaipublicfiles.com/fasttext/supervised-models/lid.176.bin

[4] https://github.com/moses-smt/mosesdecoder

[5] https://github.com/pytorch/fairseq

[6] https://github.com/rsennrich/subword-nmt.

Matrix	In Oregon , planners are experimenting with giving drivers different choices .
$r = 1$	**Dans** Oregon , planners are experimenting with giving drivers different choices .
$r = 2$	**Dans** Oregon , **les planificateurs** are experimenting with giving drivers different choices .
$r = 3$	**Dans** Oregon , **les planificateurs** are experimenting **en offrant aux** drivers different choices .
Embedded	Dans l'Orégon, les planificateurs tentent l'expérience en offrant aux automobilistes différents choix.

Figure 1: Examples of generated CSW sentences when taking English as the matrix language and varying the number r of replacements of embedded French segments (in boldface).

end, similar to a multilingual NMT model (Johnson et al., 2017), we add a tag at the beginning of each CSW sentence to specify the desired target language. Taking En-Fr as an example, we add a <EN> tag for CSW-En and a <FR> tag for CSW-Fr. We use the combination of CSW-En and CSW-Fr data for training, which implies that each source side (CSW sentence) is duplicated in the training data, once for each possible output.

- the `joint-csw` setting, which extends `multi-csw` by using one encoder and two separate decoders and training the two output languages simultaneously *with a combined loss function*: for each training (CSW) instance, the loss function sums the two prediction terms for the embedded and the matrix language. The training data remains the same.

Note that all our `Code-Switch` systems also have the ability to translate monolingual source data, in either direction.

For comparison purposes, we also use our parallel data to train two baselines: (a) regular NMT systems for the considered language pairs (`base`), similar to `base-csw`; (b) *bilingual NMT systems*, capable of translating from and into both two languages (`bilingual`). The selection of the desired target language relies on the same tagging mechanism as `multi-csw`, which means that both types of models see exactly the same examples. All resulting baseline Transformer models have the exact same hyperparameters and use the same training scheme as `Code-Switch`. Performance is computed with SacreBLEU (Post, 2018) and METEOR (Denkowski and Lavie, 2014).

3 Machine translation experiments

3.1 Results

We run tests using artificial CSW datasets, as mentioned in Section 2.2, as well as on the original test sets, in order to evaluate our models' ability to translate both CSW and monolingual sentences. Results are in Table 1 where we also separately report scores for the 'Matrix' and 'Embedded' part of the test sets. As is obvious on the `copy` line, the 'Embedded' part contains mostly source language, and corresponds to an actual translation task whereas the 'Matrix' part mostly contains target words on the source side, and is much easier to translate.

On the left part of this table, we see that the baseline systems, either with two (`base`) or one single (`bilingual`) model(s), do better on monolingual test sets than their counterparts trained on CSW data (respectively `base-csw` and `multi-csw`). For both language pairs, the observed differences are in the range of 1-1.5 BLEU points. Conversely, when translating CSW sentences, `*-csw` models perform significantly better than the corresponding baselines models, which have never seen CSW in the source.

Moreover, we note the marked differences between BLEU scores obtained by these models when the matrix language for the CSW source is the target and when the embedded language is the target. In the former case, translation is near perfect; in the latter case they nonetheless use the little information available to improve over the monolingual scores (about 1-1.5 BLEU points), nearly matching the performance of the baseline systems. This is illustrated for Fr-En, for which `joint-csw` improved from 33.7 to 35.0; in the same condition, the `bilingual` system only improves by 0.1 point.

Among the three `Code-Switch` models, `multi-csw` is the weakest, while the other two achieve comparable performance. Interestingly, with joint training (`joint-csw`), we can recover with one single system the performance of the two separate systems used in the `base-csw` condition. On the monolingual tests, this system also matches the performance of the multilingual baseline (`bilingual`), which makes it overall our best contender of the lot.

Testset	newstest2013				csw-newstest2013			
Direction	En-Es		Es-En		CSW-Es		CSW-En	
Metrics	B	M	B	M	B	M	B	M
`copy`	-	-	-	-	50.3 2.9 93.5	57.8	46.8 3.0 93.3	31.7
`base`	**33.2** 33.1 -	**58.3**	**33.8** 34.0 -	**36.4**	38.9 32.5 43.4	59.1	57.3 34.6 78.7	44.4
`bilingual`	31.9 31.9 -	57.3	32.6 32.9 -	35.9	23.3 32.3 14.5	42.0	44.2 33.3 54.5	37.5
`base-csw`	32.0 31.8 -	57.4	32.7 33.0 -	36.0	66.8 33.1 97.1	**79.8**	**66.5** 34.5 97.5	**49.4**
`multi-csw`	31.1 30.9 -	56.7	31.5 31.9 -	35.4	66.5 32.2 97.2	79.5	64.7 33.1 95.1	48.6
`joint-csw`	31.9 32.0 -	57.3	32.6 32.8 -	36.0	**66.9** 33.2 97.2	79.7	66.4 34.2 97.5	**49.4**
Testset	newstest2014				csw-newstest2014			
Direction	En-Fr		Fr-En		CSW-Fr		CSW-En	
Metrics	B	M	B	M	B	M	B	M
`copy`	-	-	-	-	50.0 2.9 93.8	55.7	46.5 2.9 93.4	33.1
`base`	**37.9** 37.7 -	**60.9**	**35.4** 35.3 -	**37.9**	45.1 37.8 52.0	64.4	61.3 36.0 84.6	47.3
`bilingual`	36.3 36.4 -	59.6	34.5 34.6 -	37.6	54.8 36.8 71.7	71.3	56.5 34.7 76.6	45.8
`base-csw`	36.7 36.7 -	59.9	34.3 34.2 -	37.5	**67.5** 37.8 95.2	**79.9**	**67.9** 35.6 97.4	**50.5**
`multi-csw`	35.2 35.3 -	58.7	32.9 32.6 -	36.8	66.7 36.3 95.1	79.5	65.8 33.7 94.6	49.4
`joint-csw`	36.2 36.2 -	59.5	34.0 33.7 -	37.3	67.4 37.3 95.4	79.8	67.7 35.0 97.4	50.3

Table 1: Translating monolingual newstest data and artificial `csw-newstest` data for two language pairs where performance is measured via the BLEU (B) and METEOR (M) scores. We also report a trivial baseline that just recopies the source text. Small numbers contain BLEU scores computed separately when the target language is the embedded language (left) and the matrix language (right). For the monolingual tests (left part), these correspond to scores computed on the same sentences that are also included in the CSW tests.

3.2 Analysis

3.2.1 Code-Switching effect

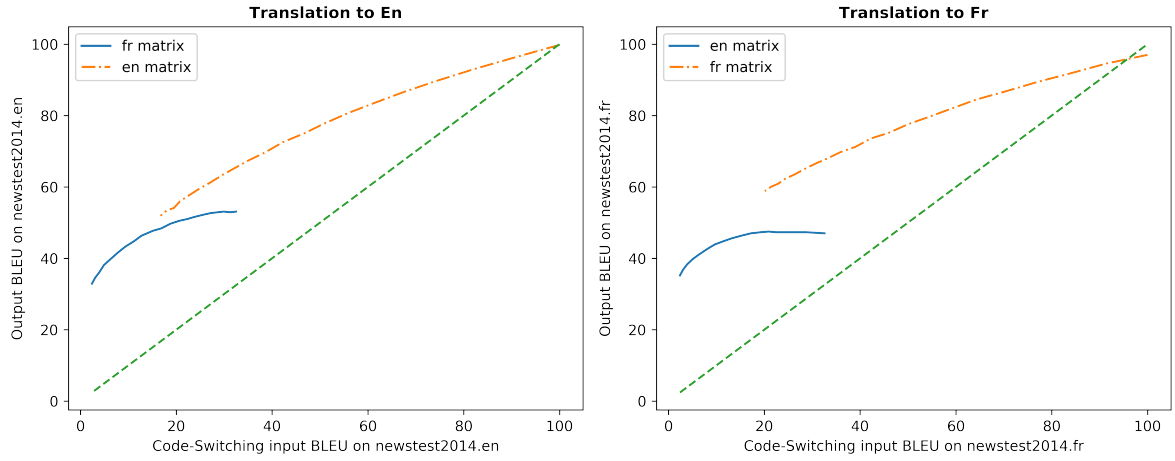
In order to better study the effect of mixing languages, we modify the synthetic data generation method to keep one language as the matrix language, in which segments are incrementally replaced by translations of the embedded language. We relax the constraint on the maximum number of replacements and generate new test sets with an increasing number of replacements, ranging from 1 to 20, resulting in 20[7] versions of the CSW test sets (in each direction). In Figure 2, we plot the BLEU scores of both source CSW sentences and their translations for En-Fr language pair, using each language as the matrix language, to visualize the impact of progressively introducing more target fragments into the source.

[7]For sentences that could not accommodate 20 replacements, we performed as many replacements as possible.

The same behavior is observed for both language pairs and directions: on average, inserting random target fragments boosts the translation performance, with a larger payoff for the first few target segments. There exists an important gap for the output BLEU scores when CSW source sentences with different matrix languages reach the same (input) BLEU scores. Even though we generate a large number of replacements, the basic grammar structure of the matrix language is still maintained. Therefore, taking the target language as matrix gives the model a pre-translated sentence structure that is much easier to reproduce.

3.2.2 Implicit LID in translation

A second question concerns the ability of the translation system to identify target fragments in the source and to copy them in the target, even though these fragments are indistinguishable from genuine source segments. We use labels computed



(a) (b)

Figure 2: Evolution of the BLEU score of source CSW data and their target translation for En-Fr. (a) Direction CSW-En. The solid curve takes Fr as the matrix language, where we progressively inject more En segments; for the dash dot curve, En is the matrix language, with a growing number of Fr segments. (b) Direction CSW-Fr. Note that the target BLEU is always much higher than the source BLEU, with about a 20 points difference. The gap between the dash dot and solid curves is due to the basic sentence structure of the matrix language (see Section 3.2.1). As dash dot curves represent insertion in the *reference target* sentence, the corresponding BLEU score is always higher than the solid curve and actually reaches 100 (in the absence of any embedded language).

during the CSW generation procedure to sort out pre-translated (target) segments from actual source segments to be translated. For instance, when translating into French, only tokens with a label `eng`, denoting English, are expected to be translated. All other tokens correspond to French words are expected to be copied. As reported in Table 2, our translation models are able to copy almost all pre-translated tokens for both language pairs and directions.

Refining the analysis, we also study whether the relative order of target words changes, or is preserved, during the translation. Table 3 reports the percentage of exact and switched-order copies. We observe again large differences with respect to the position of the matrix language. When the matrix language is the target language, the model always preserves the observed token order since it indicates a correct sentence structure for the hypothesis. When translating into the embedded language, we observe a larger number of word order changes: in this case, inserted target segments may not appear in their correct order in the CSW sentence, an issue that the model tries to fix. An example of this is in Figure 3, where we observe a swap between the input ("différent choix") and output ("choix différent") word orders.

Conversely, it is also interesting to look at the proportion of mixed language generated on the target side. Recall that in our training, the source is mixed-language, while the target is always monolingual. We use an in-house token-level language

Testset	csw-newstest2014		csw-newstest2013	
Direction	CSW-En	CSW-Fr	CSW-En	CSW-Es
to copy	42148	47337	37653	41053
copied	41567	46229	37421	40638
copy rate (%)	98.6	97.7	99.4	99.0
CSW rate (%)	0.13	0.30	0.16	0.23

Table 2: Analyzing the recopy of tokens on `csw-newstest2014` for En-Fr and `csw-newstest2013` for En-Es. We report the number of (pre-translated) tokens that should be copied, and the corresponding ratios.

	En-Fr		En-Es	
Direction	Copy	Copy+Swap	Copy	Copy+Swap
CSW-En	87.1	4.5	90.7	5.2
Mat En	97.6	0.1	98.2	0.2
Mat For	61.4	15.2	72.6	17.3
CSW-For	77.4	5.5	88.5	3.7
Mat For	84.9	0.1	97.1	0.2
Mat En	59.4	18.5	65. 6	13.2

Table 3: Percentage of sentences for which all target words have been exactly copied without and with order changes, for `csw-newstest2014` (En-Fr) and `csw-newstest2013` (En-Es). We separately report numbers for the case where the foreign language (French or Spanish) is the embedded (Mat En) or matrix (Mat For) language.

identification (LID) model to identify the language of output tokens and to detect the CSW rate on the target side. As indicated in Table 2, our models generate almost pure monolingual translations, with a very low rate of CSW text. CSW-translation models thus seem to perform some language identification, as they almost perfectly sort out target language tokens (which are almost always copied) from the source language tokens (which are always translated).

A last issue concerns morphological errors: when inserting foreign words into a matrix source, one cannot expect to always also introduce the right inflection marks, some of which can only be determined once the target context is known. Another interesting phenomenon, that we do not simulate here, is when the embedded (target) lemma is adapted bears a morphological mark that only exist in the matrix language, which means that two linguistic systems are mixed within the same word, thereby posing more extreme difficulties for MT (Manandise and Gdaniec, 2011).

To illustrate the ability to correct grammar errors in input fragments, we manually noise a CSW sentence and display its translation in Figure 3. Where the input just contains the lemma of the French word "tenter" (*to try*), the model inserts a modal "doivent" to fix the context. Another illustration is for the adjective "différent" which is moved into post-nominal position, and for which an article ("un") is inserted. This indicates that the model not only copies what already exists but also tends to adjust translations whenever necessary.

4 Computing translations in context

In this section, we evaluate CSW translation for the SemEval 2014 Task 5: L2 Writing Assistant (van Gompel et al., 2014), which can be handled as an MT task from mixed data.

4.1 Method

This task consists in translating L1 fragments in an L2 context, where the test set design is such that there is exactly one L1 insert in each utterance. We evaluated on two L1-L2 pairs: English-Spanish and French-English, and list below example test segments provided by the organizers for these pairs of languages (the insert and reference segments are in boldface):

- Input (L1=English,L2=Spanish): *"Todo ello, **in accordance** con los principios que siempre-hemos apoyado."*
  Output: *"Todo ello, **de conformidad** con los principios que siempre hemos apoy-ado."*

- Input (L1=French,L2=English): *"I **rentre à la maison** because I am tired."*
  Output: *"I **return home** because I am tired."*

The official metric for the SemEval evaluation is a word-based accuracy of the translations of the L1 fragment, which means that the L2 context of each sentence is not taken into account in scoring. Since our systems are full-fledged NMT systems, their output may not contain the reference L2 prefix and suffix. Therefore, two options are explored to compute these scores. The first is to post-process the output HYP and align it with the L2 reference context in REF. This alignment allows us to only score the relevant fragment in HYP. We refer to this option as free-dec.

The second option is to ensure that the L2 context will be present in the output translation. To this end, we use the *force decoding mode* of fairseq, implementing the methods of Post and Vilar (2018); Hu et al. (2019). We explored two different ways to express the L2 context as decoding constraints. The first turns every token in the L2 context as a separate constraint (token-cst). Continuing the previous example, "*I, because, I, am, tired.*" yield 5 constraints. The second uses the prefix and suffix of the L2 context as two multi-word constraints (presuf-cst). In this case, "*I*" and "*because I am tired.*" yield just 2 constraints. In both cases, constraints are required to be present in the prescribed order in the output.

4.2 Results

Scores are computed with the SemEval evaluation tool,[8] which enables a comparison with other submissions for this task. Results are in Table 4 and 5. For En-Es, our CSW translator outperforms the best system in the official evaluation (van Gompel et al., 2014). Note that this model is not specifically designed nor tuned in any way for the SemEval task. For Fr-En, our system achieves better performance than the forth best participating system, with a clear gap with respect to the top results. In both cases, constraint decoding hurts performance: given that the automatic copy of target segments is already nearly perfect, introducing more constraints during

[8] https://github.com/proycon/semeval2014task5

En	In Oregon , planners are experimenting with giving drivers different choices.
Fr	Dans l'Orégon, les planificateurs tentent l'expérience en offrant aux automobilistes différents choix.
CSW	In l'Oregon , planners **tentent l' expérience** with giving **automobilistes différents choix**.
Hyp	*Dans l'Orégon , les planificateurs tentent l'expérience de donner aux automobilistes différents choix.*
Noisy CSW	In l' Oregon , planners **tenter** l'expérience with giving **automobilist différent** choix.
Hyp	*Dans l'Orégon , les planificateurs **doivent tenter** l'expérience de donner **à l' automobiliste un choix différent**.*

Figure 3: A noisy Code-Switched sentence with French as both the matrix and target language.

the search has here a clear detrimental effect for this task.

	Accuracy	Word Accuracy	Recall
UEdin-run2	0.755	0.827	1.0
UEdin-run1	0.753	0.827	1.0
UEdin-run3	0.745	0.820	1.0
`multi-csw`			
`free-dec`	0.755	0.827	1.0
`token-cst`	0.749	0.824	1.0
`presuf-cst`	0.751	0.827	1.0
`joint-csw`			
`free-dec`	0.773	0.842	1.0

Table 4: Results of SemEval 2014 Task 5 for En-Es.

	Accuracy	Word Accuracy	Recall
UEdin-run1	0.733	0.824	1.0
UEdin-run2	0.731	0.821	1.0
UEdin-run3	0.723	0.816	1.0
CNRC-run1	0.556	0.694	1.0
`multi-csw`			
`free-dec`	0.554	0.685	0.996
`token-cst`	0.531	0.665	0.990
`presuf-cst`	0.519	0.658	0.982
`joint-csw`			
`free-dec`	0.626	0.744	0.994

Table 5: Results of SemEval 2014 Task 5 for Fr-En.

To better study the performance gap between these language pairs, we additionally score the development and test data with BLEU and METEOR. Results in Table 6 show that for these metrics, we achieve performance that are in that same ballpark for the two language pairs, suggesting that the observed difference in the SemEval metric is likely due to a mismatch between references and system outputs. The official metric is a word accuracy which may exclude acceptable translations by exact token match.

	`multi-csw`		`joint-csw`	
Dataset	B	M	B	M
Fr-En dev	97.3	75.6	97.6	76.4
Fr-En test	90.1	64.1	91.0	66.1
En-Es dev	97.4	98.8	97.6	99.0
En-Es test	89.9	95.3	90.4	95.5

Table 6: Results of other metrics on SemEval data. METEOR scores for the Fr-En SemEval test are much worse than for En-Es. This is mostly due to the high "fragmentation penalty" computed by METEOR for English; the corresponding average F_{mean} is about 0.99, showing that translations are mostly correct.

5 Related work

Research in the area of NLP for CSW has mostly focused on CSW Language Modeling, especially for Automatic Speech Recognition (Pratapa et al., 2018; Garg et al., 2018; Gonen and Goldberg, 2019; Winata et al., 2019; Lee and Li, 2020). Evaluation tasks, benchmarks have also been prepared for LID in user generated CSW content (Zubiaga et al., 2016; Molina et al., 2016), Named Entity Recognition (Aguilar et al., 2018), Part-of-Speech tagging (Ball and Garrette, 2018; Aguilar et al., 2020; Khanuja et al., 2020) and Sentiment Analysis (Patwa et al., 2020). CSW was also found useful in foreign language teaching: Renduchintala et al. (2019a,b) showed that replacing words by their counterparts in foreign language helps to learn foreign language vocabulary.

Regarding MT, most past work has focused on using artificial CSW data to help conventional translation systems. Huang and Yates (2014) used CSW corpus to improve word alignment and statistical MT. Dinu et al. (2019) experienced replacing and concatenating source terminology constraints by the corresponding translation(s) to boost the accuracy of term translations. Song et al. (2019a) shared the same idea by replacing phrases with prespecified translation to perform "soft" constraint decoding. A different line of research is in (Bulte and Tezcan, 2019; Xu et al., 2020; Pham et al., 2020), who explore ways to combine a source sentence with similar translations extracted from translation memories. Yang et al. (2020) also pretrained translation models by predicting original source segments from generated CSW sentences and claimed better results compared to other pre-

training methods (Conneau and Lample, 2019; Song et al., 2019b). Nevertheless, there barely exists work aimed at translating CSW sentences. Johnson et al. (2017) mentioned using a multilingual NMT system to translate CSW sentence to a third target language by showing only one example. To the best of our knowledge, only one parallel Arabic-English CSW corpus was specifically released for MT applications (Menacer et al., 2019). This CSW data was extracted from the UN data with Arabic as the matrix language: while translations into English were readily available, the purely Arabic side of the corpus was obtained using Google Translate to fill the missing Arabic bits.

6 Conclusion and outlook

In this study, we present a data augmentation method to generate artificial CSW data. We have shown that artificial data generated could be used to train NMT systems to translate both monolingual and CSW sentences (in one or even two different languages). With joint training of the two languages, we were able to build systems that were as good as a baseline bilingual system on monolingual texts, and much better for CSW texts. Our system does not need any explicit language identification and almost perfectly sorts out source tokens from target tokens in a CSW utterance. Another interesting feature of our system is that it always output monolingual translations. We finally report state-of-the-art results for the SemEval L2 Writing Assistant task for Es-En, while the related results for Fr-En are still somewhat lagging behind the best scores.

In the future, we would like to generate more realistic CSW data from monolingual sentences using a translation model. We also plan to explore ways to translate CSW texts simultaneously into both languages, so that the two decoding processes can mutually influence one another: in a first step in that direction, we have shown that training with a joint loss was actually beneficial for the translation into the two languages. Another line of research would be to continue experimenting with realistic language data, also containing other phenomena such as morphological binding. Finally, we also intend to study the somewhat more realistic condition where a mixture of languages A and B is translated into language C; we believe that the artificial CSW generation methods developed in our work would also be effective for this task.

7 Acknowledgements

This work was granted access to the HPC resources of IDRIS under the allocation 2021-[AD011011580R1] made by GENCI. The authors wish to thank Josep Crego for his comments of an earlier version of this work. We also would like to thank the anonymous reviewers for their valuable suggestions. The first author is partly funded by Systran and by a grant from Région Ile-de-France.

References

Gustavo Aguilar, Fahad AlGhamdi, Victor Soto, Mona Diab, Julia Hirschberg, and Thamar Solorio. 2018. Named entity recognition on code-switched data: Overview of the CALCS 2018 shared task. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 138–147, Melbourne, Australia. Association for Computational Linguistics.

Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. 2020. LinCE: A Centralized Benchmark for Linguistic Code-switching Evaluation. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 1803–1813, Marseille, France. European Language Resources Association.

Gustavo Aguilar and Thamar Solorio. 2020. From English to code-switching: Transfer learning with strong morphological clues. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8033–8044, Online. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kalika Bali, Jatin Sharma, Monojit Choudhury, and Yogarshi Vyas. 2014. "I am borrowing ya mixing ?" an analysis of English-Hindi code mixing in Facebook. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 116–126, Doha, Qatar. Association for Computational Linguistics.

Kelsey Ball and Dan Garrette. 2018. Part-of-speech tagging for code-switched, transliterated texts without explicit language identification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3084–3089, Brussels, Belgium. Association for Computational Linguistics.

Hedi M Belazi, Edward J Rubin, and Almeida Jacqueline Toribio. 1994. Code switching and x-bar theory: The functional head constraint. *Linguistic inquiry*, pages 221–237.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bram Bulte and Arda Tezcan. 2019. Neural fuzzy repair: Integrating fuzzy matches into neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1800–1809, Florence, Italy. Association for Computational Linguistics.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32, pages 7059–7069. Curran Associates, Inc.

Josep M. Crego, José B. Mariño, and Adrià De Gispert. 2005. Reordered search, and tuple unfolding for Ngram-based SMT. In *In Proceedings of the MT Summit X*, pages 283–289.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, Baltimore, Maryland, USA. Association for Computational Linguistics.

Georgiana Dinu, Prashant Mathur, Marcello Federico, and Yaser Al-Onaizan. 2019. Training neural machine translation to apply terminology constraints. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3063–3068, Florence, Italy. Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 866–875. Association for Computational Linguistics.

Saurabh Garg, Tanmay Parekh, and Preethi Jyothi. 2018. Code-switched language models using dual RNNs and same-source pretraining. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3078–3083, Brussels, Belgium. Association for Computational Linguistics.

Hila Gonen and Yoav Goldberg. 2019. Language modeling for code-switching: Evaluation, integration of monolingual data, and discriminative training. 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 4175–4185, Hong Kong, China. Association for Computational Linguistics.

J. Edward Hu, Huda Khayrallah, Ryan Culkin, Patrick Xia, Tongfei Chen, Matt Post, and Benjamin Van Durme. 2019. Improved lexically constrained decoding for translation and monolingual rewriting. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 839–850, Minneapolis, Minnesota. Association for Computational Linguistics.

Fei Huang and Alexander Yates. 2014. Improving word alignment using linguistic code switching data. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1–9, Gothenburg, Sweden. Association for Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Simran Khanuja, Sandipan Dandapat, Anirudh Srinivasan, Sunayana Sitaram, and Monojit Choudhury. 2020. GLUECoS: An evaluation benchmark for code-switched NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3575–3585, Online. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Grandee Lee and Haizhou Li. 2020. Modeling code-switch languages using bilingual parallel corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 860–870, Online. Association for Computational Linguistics.

Esmé Manandise and Claudia Gdaniec. 2011. Morphology to the rescue redux: Resolving borrowings and code-mixing in machine translation. In *Systems and Frameworks for Computational Morphology*, pages 86–97, Berlin, Heidelberg. Springer Berlin Heidelberg.

Mohamed Menacer, David Langlois, Denis Jouvet, Dominique Fohr, Odile Mella, and Kamel Smaïli. 2019.

Machine Translation on a parallel Code-Switched Corpus. In *Canadian AI 2019 - 32nd Conference on Canadian Artificial Intelligence*, Lecture Notes in Artificial Intelligence, Ontario, Canada.

Giovanni Molina, Fahad AlGhamdi, Mahmoud Ghoneim, Abdelati Hawwari, Nicolas Rey-Villamizar, Mona Diab, and Thamar Solorio. 2016. Overview for the second shared task on language identification in code-switched data. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 40–49, Austin, Texas. Association for Computational Linguistics.

Carol Myers-Scotton. 1997. *Duelling languages: Grammatical structure in codeswitching*. Oxford University Press.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Parth Patwa, Gustavo Aguilar, Sudipta Kar, Suraj Pandey, Srinivas PYKL, Björn Gambäck, Tanmoy Chakraborty, Thamar Solorio, and Amitava Das. 2020. SemEval-2020 task 9: Overview of sentiment analysis of code-mixed tweets. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 774–790, Barcelona (online). International Committee for Computational Linguistics.

Carol W Pfaff. 1979. Constraints on language mixing: Intrasentential code-switching and borrowing in Spanish/English. *Language*, pages 291–318.

Minh Quang Pham, Jitao Xu, Josep Crego, François Yvon, and Jean Senellart. 2020. Priming neural machine translation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 462–473, Online. Association for Computational Linguistics.

Shana Poplack. 1978. *Syntactic structure and social function of code-switching*, volume 2. Centro de Estudios Puertorriqueños, City University of New York].

Shana Poplack. 1980. Sometimes i'll start a sentence in spanish y termino en espaÑol: toward a typology of code-switching 1. *Linguistics*, 18:581–618.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1314–1324, New Orleans, Louisiana. Association for Computational Linguistics.

Adithya Pratapa, Gayatri Bhat, Monojit Choudhury, Sunayana Sitaram, Sandipan Dandapat, and Kalika Bali. 2018. Language modeling for code-mixing: The role of linguistic theory based synthetic data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1543–1553, Melbourne, Australia. Association for Computational Linguistics.

Adithya Renduchintala, Philipp Koehn, and Jason Eisner. 2019a. Simple construction of mixed-language texts for vocabulary learning. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 369–379, Florence, Italy. Association for Computational Linguistics.

Adithya Renduchintala, Philipp Koehn, and Jason Eisner. 2019b. Spelling-aware construction of macaronic texts for teaching foreign-language vocabulary. 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 6438–6443, Hong Kong, China. Association for Computational Linguistics.

Shruti Rijhwani, Royal Sequiera, Monojit Choudhury, Kalika Bali, and Chandra Shekhar Maddila. 2017. Estimating code-switching on Twitter with a novel generalized word-level language detection technique. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1971–1982, Vancouver, Canada. Association for Computational Linguistics.

Younes Samih, Suraj Maharjan, Mohammed Attia, Laura Kallmeyer, and Thamar Solorio. 2016. Multilingual code-switching identification via LSTM recurrent neural networks. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 50–59, Austin, Texas. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Sunayana Sitaram, Khyathi Raghavi Chandu, Sai Krishna Rallabandi, and Alan W. Black. 2019. A survey of code-switched speech and language processing. *CoRR*, abs/1904.00784.

Kai Song, Yue Zhang, Heng Yu, Weihua Luo, Kun Wang, and Min Zhang. 2019a. Code-switching for enhancing NMT with pre-specified translation. In

Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 449–459, Minneapolis, Minnesota. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019b. Mass: Masked sequence to sequence pre-training for language generation. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 5926–5936. PMLR.

Maarten van Gompel, Iris Hendrickx, Antal van den Bosch, Els Lefever, and Véronique Hoste. 2014. SemEval 2014 task 5 - L2 writing assistant. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 36–44, Dublin, Ireland. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Genta Indra Winata, Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2019. Code-switched language models using neural based synthetic data from parallel sentences. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 271–280, Hong Kong, China. Association for Computational Linguistics.

Jitao Xu, Josep Crego, and Jean Senellart. 2020. Boosting neural machine translation with similar translations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1580–1590, Online. Association for Computational Linguistics.

Zhen Yang, Bojie Hu, Ambyera Han, Shen Huang, and Qi Ju. 2020. CSP: Code-switching pre-training for neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2624–2636, Online. Association for Computational Linguistics.

Arkaitz Zubiaga, Iñaki San Vicente, Pablo Gamallo, José Ramom Pichel, Iñaki Alegria, Nora Aranberri, Aitzol Ezeiza, and Víctor Fresno. 2016. TweetLID: a benchmark for tweet language identification. *Language Resources and Evaluation*, 50(4):729–766.