GROUCH: Generative Recurrent translations Of Under-resourced Conversations with a Hybrid model

Chaitanya Ravuri
MIT
cravuri@mit.edu

Karissa Sanchez
MIT
ksanchez@mit.edu

Chanwoo Yoon
MIT
chanwooy@mit.edu

Abstract

There are many under-resourced languages with sparse datasets and few expert translators. Collecting data for these languages on the scale that neural network based models need is expensive and time-consuming. For this project, we are mimicking a low-resource language by using a small dataset in Spanish. We want to test if using a hybrid system with Neural Machine Translation (NMT), appending words in the target language using a dictionary, and utilizing a trained grammar correction model provide better speed and accuracy on under-resourced datasets than just a pure NMT model with more data. Our hybrid model consists of a precomputation step, the NMT model, and a postcomputation step. The precomputation includes appending either [UNK] or the translated word to the end of the input sentence for each word in the sentence, depending on if the word has a translation in the dictionary we used. The main model is an encoder decoder model which optionally includes teacher forcing. The post computation is utilizing a pretrained grammar correction model to improve the quality of the English output of the model. We analyzed which combination of features will produce the optimal output measured by BLEU score, and found that using the pretrained grammar correction model and teacher forcing allows for the best translations.

1 Introduction

Neural machine translation (NMT) has achieved state-of-the-art performance when trained on languages with large parallel datasets. Many NMT models are end-to-end, training to translate directly from the source sentence to the target sentence. In such a system, the model needs to learn a great deal about the source and target languages, in addition to learning how to translate between them. This learning is possible with a large training dataset, but for many languages, datasets at this scale simply do not exist. For these under-resourced languages, collecting large amounts of translation data is expensive and time-consuming due to a lack of expert translators.

In this paper, we wish to investigate a number of methods to improve the performance of NMT in the domain of under-resourced languages. Specifically, we will attempt various manipulations of both the input and output of the model, adding useful information that can increase the accuracy of the model.

The specific goal of our project is to create a system which can translate from a low to a high resource language. We decided to work with Spanish instead of a true low-resource language for two main reasons. The first reason is that it allows us more flexibility in our experiments. For instance, we use a Spanish-English dictionary of about 4,000 words in our precomputation. Not all low-resource languages may have such online resources available, but since Spanish does, we can utilize it to see if building an extensive dictionary for a low-resource language is a valuable use of time. The second reason is that working with an under-resourced language entails working with the community of its speakers. Due to our limited time constraints for this project, we could not establish such a connection and did not want to determine on behalf of a community which we are not part of how they could most benefit from our model.

The problem with an end-to-end system is that it assumes very little about the source and target languages, so it cannot take advantage of any knowledge that we have about both languages. One key piece of information we have is a dictionary. Although an under-resourced language may not have many translated sentence pairs, at the very least, a dictionary of individually translated words would exist. If we augment our model so that this dictionary is already embedded in it, then the model

would not have to learn this mapping, reducing the amount of data it needs.

Another potential augmentation is on the output. A significant use case of machine translation is to translate from an under-resourced language to a more widely spoken language, like English. There exist large amounts of data in English, and we can take advantage of this data to augment our NMT model. For example, we can train a model that corrects grammatical errors in English sentences. Then, we can pass the output of the NMT model through this grammar correction model to get our translation. Our correction model will fix any errors the NMT model makes, so the NMT model will not have to have a full understanding of English grammar, hence reducing the amount of data it needs.

2 Related Work

Bonet et. al. used a similar idea to ours, with a NMT model feeding into a rule-based system that edited the output of the NMT model. However, their goal was not to improve translation for under-resourced languages but to just improve the accuracy of translation in general, and they did not use modern NMT architectures like transformers. We will be improving on their implementation and testing the accuracy in the under-resourced domain. (Bonet et al., 2011)

Both Torregrosa et. al. and Stajner et. al. also proposed similar models as ours. In both these papers, the authors described a model which transformed the input before feeding it into a translation model. Torregrosa et. al. used rules to add part of speech tags to the input text that would make it easier for the NMT model to translate (Torregrosa et al., 2019), and Stajner et. al. used a neural-based text simplification model as a preprocessing step to make the translation easier (Štajner and Popović, 2019).

In 2016, Zhang et. al. explored the use of character-level convolutional networks (ConvNets) for text classification and showed that ConvNets could achieve state-of-the-art or similar results. While this topic is not directly related to our research interests, their methodology of expanding their training data by data augmentation—specifically the replacement of some of the words with synonyms from a thesaurus to generate valid training sentences—gives us some guidance in overcoming the size limitation of under-resourced language datasets. Their reported data suggests a moderate increase in precision from using data augmentation. With the additional fact that adding variants of a sentence can help to generalize the trained model, we will consider this instance of data augmentation as a mean of improving our model's translations (Zhang et al., 2015).

In 2015, Gulcehre et. al. noticed that huge success in neural machine translation (NMT) was mainly due to large and well formatted datasets. They then tried to find if additionally utilizing information from monolingual corpora can be helpful—especially when large and well formatted data may not be available. They built a language model from monolingual corpora and incorporated it into the model training process. The process of incorporating was done by having the language model give feedbacks—by scoring next word candidates based on the weighted sum of the scores by the translation model and the language model—and concatenating the hidden states of the language model and the hidden states of the hidden model. For our research, we will borrow the concept of leveraging monolingual corpora—which is more so important as our data is under-resourced—and the concept of concatenating hidden states to give more expressibility to our model (Gulcehre et al., 2015).

In 2018, Zhang and Yang explored the effects of applying small perturbations to word embeddings for the task of sentence classification. They applied gaussian noise, bernoulli noise, and adversarial training individually and noticed various yet positive growths over all datasets. While their goal was to address the common problem of overfitting in neural networks, their work is still relevant to our research interests in machine translation as adding noise to word embeddings results in more data to train on. With our constraint being under-resourced corpora, which is more prone to overfitting, we will consider this paper's approach as one of the methods to generalize and make use out of data as much as possible (Zhang and Yang, 2018).

3 Models

As we explore which methods are most effective for machine translation of an under-resourced language, there will be three domains in which we will add variation into our model. These domains are the manipulation to the inputs, changes to the core model, and manipulation to the outputs. How

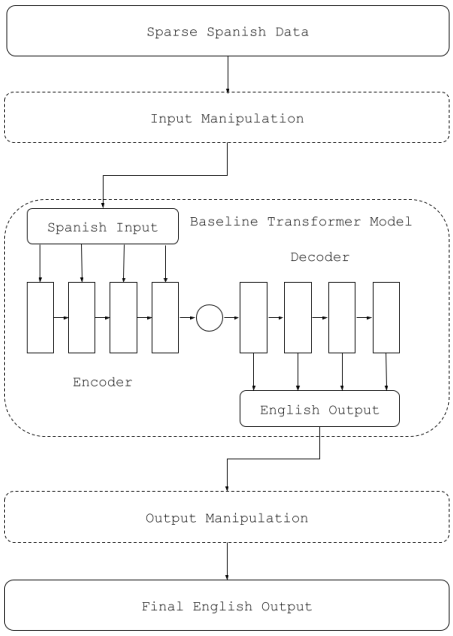

Figure 1: NMT Model with potential manipulation of the input and output sentences.

these different steps are integrated into the whole system is shown through the dashed boxes in Fig. 1.

The input of the model is a sequence of tokens in the source language: $x = (x_1, \ldots, x_n)$. At the input manipulation step, we apply some transformation function f to augment the input with useful information by appending values. The output of that step is $f(x) = x' = (x_1, \ldots, x_n, x'_1, \ldots, x'_n)$.

The core translation model takes in x' and produces a series of tokens in the target language $y' = (y'_1, \ldots, y'_m)$. The output manipulation step applies another transformation function g to the output of the previous step, producing the final translation $g(y') = y = (y_1, \ldots, y_m)$.

Our paper will explore various input manipulation functions f, output manipulation functions g, and core models in an attempt to determine which overall model works best for an under-resourced language.

3.1 Input Manipulation

Our model's method for manipulating the input for the model is using some form of "manual" translation from Spanish to English words in the input sentence. In the case of our model, this means leveraging a Spanish to English Dictionary with word to word mappings, and using these mappings to append English words to the end of the Spanish input sentence before passing it to the core model. The input which is passed to the core model will be twice as large as the original input, because for each word in the input, either the dictionary's translation will be appended to the input sentence, or an [UNK] token will be appended if the dictionary does not contain the input word. For words in Spanish which have multiple English translations, we will choose the one which was first to appear in the XML file. In future projects, it would be useful determining a more effective manner to handle the case of multiple possible translations. This step could be useful with processing an under-resourced language because a dictionary is more attainable than an extensive parallel corpus. Through this step, we are preprocessing the data as a method to ensure that the semantic meaning of the sentence is preserved as much as possible. Through further experiments with this preprocessing step, we will analyze its effect on the quality of retaining the original meanings of translated sentences.

3.2 Core Model Manipulation

One method for altering the core model for our translation system is using teacher forcing for the decoder step of the transformer. Teacher forcing is a feasible customization that could be made for a model which is training an under-resourced language, since this method involves using the correct outputs from the target language and these should already be available with a small parallel corpus. We are hoping that by implementing teacher forcing in the decoder of the transformer, the model will be more accurate and the hidden layers cannot stray too far from the correct output and propagate its own errors.

3.3 Output Manipulation

In order to manipulate the output for our model, we are using a pretrained model which will be fed possible erroneous English sentences and alter these sentences to have correct English grammar. This change is possible for a machine translation for an under-resourced language, as long as the translation is occurring in the direction of an under-resourced language to a well-resourced language, since a well-resourced language is much more likely to have a pretrained grammar correction model ready for use. We expect that this change will increase the accuracy of the model because it will cause the output to be more grammatically correct, as would be the expected output.

4 Experiments

In our experiments, we will attempt to determine which choice of input manipulation, core model, and output manipulation produces the best results for a low-resource dataset. Our choices are as follows:

- There will be two choices for the input manipulation method: (1) No input manipulation and (2) augmenting each word with its translation.
- There will be two choices for the core model that we use: (1) base transformer model and (2) transformer model with teacher forcing.
- There will be two choices for the output manipulation method: (1) No output manipulation and (2) pretrained grammar correction.

Overall, we will have a total of 8 possible variations of our model. We will implement and train each of these 8 variations so as to compare their results.

4.1 Dataset

We will be using the Anki Spanish-to-English dataset (https://www.manythings.org/anki/) in training our model. The full dataset has 100k pairs of Spanish and English sentences, of which we will use 80k pairs for training, 10k pairs for validation, and 10k pairs for testing.

When constructing our low-resource dataset, we will randomly sample 10% of the Anki dataset. So, our low-resource dataset will have 8k training pairs and 1k validation pairs. Each of the 8 variations of our model will be trained on this low-resource dataset. The SOTA model will be pretrained, and we will test all models using the 10k test pairs.

4.2 Experimental Details

The dictionary we used for the precomputation is the Spanish-English Dictionary provided by the FreeDict project. The XML file containing all the pairs is parsed and made into a python dictionary with Spanish words as keys and a list of English words as values, which our model uses to find words to append to the original input sentence.

We trained the baseline transformer model using the RMSProp optimizer with the learning rate of 10^{-3}. At test time, we generated output sentences using greedy decoding, stopping the generation when an "[END]" token was generated. When teacher forcing was applied, the model was trained for 30 epochs to reach convergence. Without teacher forcing, the model was trained for 50 epochs to approximately reach convergence. The training was done through cloud computational resources from Google Colab Pro+.

For our grammar correction model, we used the pretrained T5 base grammar correction model from Hugging Face (https://huggingface.co/vennify/t5-base-grammar-correction). This model generates a revised version of an input text in English. We simply added this model to the end of our translation model during test time when generating sentences.

4.3 Metric

The metric we will use to assess the accuracy of the models is the BLEU score. The BLEU score is commonly used in translation tasks and is a good measure of how close our output is to the target translation.

After training each of the 8 models, we will compute their BLEU score on the test set. The model with the highest BLEU score is the one we will assume is the best.

5 Results

The results are summarized in Table 2 and Table 3. The baseline model trained only with teacher forcing and without any input or output manipulations yields the score of **13.59**. The model from applying teacher forcing and grammar correction results in the highest BLEU score of **17.81**. Overall, we observe that applying either grammar correction or vocabulary augmentation leads to a noticeable increase in accuracy. However, applying both manipulations appears to cause conflicts between the two methods and result in non optimal performance.

When teacher forcing is not used, we observe much lower scores regardless of whether input or output manipulations are applied. The model trained after vocabulary augmentation was applied results in the highest BLEU score of 3.290. While vocabulary augmentation by itself improves the model's accuracy as it did in models trained using teacher forcing, we observe that applying grammar correction alone reduces the BLEU score significantly from 1.804 to 0.337. In fact, it appears the negative effect from grammar correction is greater than the positive effect from vocabulary augmentation: applying both manipulations results in lower

Table 1: Sample translations after training base transformer model on low-resource dataset.

Input	Model Translation
Tom no interferirá.	tom doesnt need to answer.
Los pájaros están cantando.	the bird are on singing
Mi tía tenía tres hijos.	my aunt had three children
El niño derramó la tinta, pero no fue a propósito.	the child playing the least what of europe was purpose
Tom no podía pensar en nada más.	tom could think of no one else
Ese es un clásico.	this is a a a a she a make

Table 2: BLEU Scores for each model (with teacher forcing).

	No Grammar Correction	Grammar Correction
No Vocabulary Augmentation	13.59	**17.81**
Vocabulary Augmentation	15.70	16.67

Table 3: BLEU Scores for each model (without teacher forcing).

	No Grammar Correction	Grammar Correction
No Vocabulary Augmentation	1.804	0.337
Vocabulary Augmentation	3.290	0.425

score than that of the base model trained only using teacher forcing.

6 Discussions

6.1 Limitations from the dictionary

Initially, we predicted the optimal score to come from applying both grammar correction and vocabulary augmentation. Our rationale was that since the source sentence representations are enriched from implicitly mapping each source word to the target word and since the model's outputs are post processed by pipelining them to grammar correction model, the overall model could only get better and not worse.

However, this prediction would only hold in the case of the perfect dictionary that maps each English word to the complete set of its Spanish counterpart in various conjugate forms. When we analyzed the dictionary we were using, we found many words to be lacking their full translations. Consequently, in the process of changing our source sentence embeddings, wrong conjugate forms or wrong words must have been introduced, and with the same words being in different sentences with different contexts, the model may have learned to translate different conjugate forms of a word erroneously. In turn, when the resulting translations with changed usage of a word—e.g., translated as a gerund when it was originally a participle—were fed to the grammar correction model, the error may have propagated to reduce the overall BLEU score, which agrees with the observed drop in score from applying both input and output manipulations.

6.2 Teacher forcing

As to the stark difference in scores between the models trained using teacher forcing and those that were not, we mainly attribute it to the relationship between the two factors: the size of our dataset and the limitations of not applying teacher forcing.

Specifically, not having teacher forcing is analogous to vanilla RNN: at each time step, a token is generated based on the sequence so far, and the original sequence becomes updated with this token rather than the correct token, with this process repeating until the end token is generated. The similarity between this process and that of vanilla RNN means not applying teacher forcing must result in similar issues like instability during training (e.g., exploding/vanishing gradients) and slow convergence. And in fact, this is what we observed during

the training: at 30 epochs where its counterparts that were trained using teacher forcing converged, the model nearly for all input sentences fell into degenerate cases and produced sequences solely comprising of "common" words like "the" and "a". At 50 epochs, the model avoided degenerate cases, but the sentences were mostly nonsense. At 100 epochs, the model was generally correct for the first few words for each sequence, but the words further down the sequences were mostly incorrect, and the model was far from reaching convergence. Moreover, while the training accuracy was reaching nearly 100% towards the end, the validation accuracy was plateauing at around 70% since epoch 30, showing another sign of the model's instability.

While this slow convergence may become beneficial given a big enough dataset—since even if the model's initial error is not corrected and propagates down the sequence, the bigger learning space allows errors to be identified and fixed and therefore greater generality—we are forced to use a small dataset(s) due to the constraint of translating from under-resourced languages. Hence, our model is unstable due to its potential limitations on small datasets, which agrees with the result, and not much can be done in our context, which suggests that modifying the baseline model may not be ideal in improving translations of under-resourced languages.

6.3 No teacher forcing + grammar correction/vocabulary augmentation

Another difference from not having teacher forcing is that applying grammar correction rather causes a drop in BLEU scores when applying vocabulary correction causes an increase. This is most likely due to the base transformer model trained without teacher forcing having a lackluster performance. Even if the grammar correction model tries to fix the transformer outputs, if the sequences are completely incoherent and unintelligible, the model cannot do anything to more accurately capture the true meaning and can only keep the sequences incoherent at its best.

However, vocabulary augmentation is independent of the transformer model's performance since it comes before the model in structure. And the augmentation to enrich the sentence embeddings certainly helps in model training, so we observe the increase in BLEU scores from using the augmentation.

6.4 Tradeoffs

Teacher forcing is predominant in practice due to its inherent structure that allows parallelization. In fact, the models with teacher forcing took roughly 30 seconds per epoch when the models without took roughly 4 times as much. However, due to the small size of our dataset, we inferred that training the model without teacher forcing was feasible and that it might be possible to reap potential benefits from training in sequence. While the same small dataset caused the model's instability, given slightly larger datasets, the model may avoid such problem, be trained in a reasonable amount of time, and benefit from training in sequence.

7 Conclusion

We explored many aspects in improving the machine translations of under-resourced languages. Specifically, we experimented with input manipulation (vocabulary augmentation), the base transformer model (teacher forcing), and output manipulation (grammar correction model). The overall results seem to suggest that it is more reliable and effective to leverage other larger datasets and models that are available to improve the inputs and the outputs. Additionally, the constraint of having only small datasets to train the main model limits the number and effects of changes that could be applied. For our specific context, the grammar correction as the output manipulation yields the best result, but we expect the combination of changing both the inputs and the outputs to be optimal given a better dictionary.

8 Impact Statement

The immediate impact of our work is having another way to support and revitalize nearly extinct languages whose numbers of native speakers are nearing zero (and hence low-resourced). According to UNESCO, nearly half of the 6000 languages worldwide are endangered or becoming endangered as such, so our work could then be employed to preserve these cultures.

However, there may be many concerns for our work: for the concern that our model is not fully usable yet, we want to emphasize that the model was trained on the dataset with only around 8k samples. Real life datasets are likely to be bigger, and further augmentations can be done such as substituting some words like subjects, direct objects, indirect objects with other words to artificially expand our

dataset to improve the model even more. Hence, our approach to this problem has the potential for much better performance.

Some may also wonder if input manipulations can be done from the start when a language is under-resourced to the point that sources like dictionaries are not readily available. We agree with this argument, but we also want to point out that making some sources like a dictionary is a much easier task compared to obtaining an extensive parallel corpus. After all, only the mappings between each word needs to be created, and there is no need to understand contexts between different groups of words in a sentence.

Lastly, we must consider the potential adverse effects of our model. A low-resource language model like ours can cause unintended harms to nearly extinct languages with very few speakers. For instance, suppose input and output manipulations are done, and the observed accuracy which is based on a predefined metric such as BLEU increases drastically. Yet, if the resulting model in exchange for a better BLEU score now learned to use certain parts of speech completely erroneously without native speakers to correct the errors, non-native speakers may learn the wrong language if they are relying on the output of the model to do so. In short, the model's errors could propagate to language learners and the language could become corrupted. For this issue, we emphasize that when our model's method is being used to translate these languages, users must not blindly accept the results.

References

Cristina España I Bonet, Lluís Màrquez i Villodre, Gorka Labaka, Arantza Díaz de Ilarraza Sánchez, and Kepa Mirena Sarasola Gabiola. 2011. Hybrid machine translation guided by a rule–based system. In *MTSUMMIT*.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation.

Sanja Štajner and Maja Popović. 2019. Automated text simplification as a preprocessing step for machine translation into an under-resourced language. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1141–1150, Varna, Bulgaria. INCOMA Ltd.

Daniel Torregrosa, Nivranshu Pasricha, Maraim Masoud, Bharathi Raja Chakravarthi, Juan Alonso, Noe Casas, and Mihael Arcan. 2019. Leveraging rule-based machine translation knowledge for under-resourced neural machine translation models. In *Proceedings of Machine Translation Summit XVII: Translator, Project and User Tracks*, pages 125–133, Dublin, Ireland. European Association for Machine Translation.

Dongxu Zhang and Zhichao Yang. 2018. Word embedding perturbation for sentence classification. *CoRR*, abs/1804.08166.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *CoRR*, abs/1509.01626.