Resource-Scarce Machine Translation:

Designing a System to Facilitate the Data Collection and Machine Translation of Indigenous and Resource-Scarce Languages

Author: Colton Bishop
Advisor: Srinivas Bangalore, Ph.D.

Junior Independent Work, Fall 2019

Abstract

This paper proposes and evaluates a framework intended to support communities whose language is unsupported by online translation tools. Through active learning and neural machine translation models optimized for low-resource data contexts, this system aims to empower communities to amass their own translation data (between their own language and English) and then automatically provide them with real-time online translators that they can begin to use and refine. This paper examines several different approaches to both the active learning of new translation data and to optimizing the network for generalized resource-scarcity. Experimentation shows that both the optimal active learning approach and resource-scarce optimization produce better results than the baseline: an optimized sequence-to-sequence network with attention that learns sentences randomly by reading through translated books. It further explores the implications of launching and beta-testing this platform in two very different communities: northern Australia and northern India.

Table of Contents

1 Introduction

Despite the recent advancements in machine translation in the past decades, the vast majority of indigenous languages are still not translatable online. There are three principle reasons for this reality. Firstly, traditional approaches to machine translating indigenous and low-resource languages have relied on rule-based systems in which grammar rules are at least partially hard-coded into the system.[1] However, there are often few, if any, bilingual speakers or linguists of these languages who possess the computational linguistic skills needed to parse the data and encode additional grammar rules to refine such systems. Secondly, most of these languages lack a substantial corpus of parallel data (pairs of translated sentences between the indigenous language and another language) with which to train a non-rule-based translation system[2]. Thirdly, there is limited economic incentive for collecting this data or building such systems despite the clear positive implications such systems would have in helping members of these minority communities integrate into formal educational structures, navigate legal systems, and participate more easily and fully in society.

1.1 Motivation

The end-to-end system proposed by this paper seeks to shift the power to collect and utilize data into the hands of the people who would most benefit from it. The goal is to provide a simplified, centralized web-platform that allows groups of individuals to amass

---

[1] Font Llitjós, A., & Aranovich, R. (n.d.). Building Machine translation systems for indigenous languages. *Carnegie Mellon University* . Retrieved from

[2] Ward, M. (n.d.). *CALL for Endangered Languages: Challenges and Rewards.* Retrieved from:https://www.researchgate.net/publication/248906565_CALL_for_Endangered_Languages_Challenges_and_Rewards.

their own translation data (between their own language and English) which will automatically be used to train real-time online translators that they can instantly begin to use and refine. This is certainly an ambitious goal, but this paper argues that with the proper organization and direction of volunteers and a properly designed system, something like this is feasible and could have real impact. To see this, we can imagine a small indigenous language-community of 5,000. Among this group, there might be 1,000 speakers who are bilingual (speaking, say, English and their indigenous language) and of these 2,000, there might be 100 individuals that appreciate the value that a real-time online translator could provide to their community and are thus self-incentivized to volunteer to provide an amount of daily translation labor. If each of these volunteers provides one hour of translation labor per day for five days each week, this community could give 26,000 hours of translation labor each year. If we say that on average each volunteer can translate 3 sentences per minute, and thus 120 sentences per hour, then 3,120,000 sentence pairs could be generated each year. The possibility of financial compensation for translation laborers would likely increase the quality and consistency of translators and translations, and as many governments are already given significant funding as reparations to support translation of indigenous language,[3] this sort of compensation-scheme could very realistically be put into practice.

If these sentences were thoughtfully and deliberately selected, this corpus may be sufficient to train a real-time translator that is feasible and impactful in many communities and contexts.

---

[3] Bandia, P. (2014). Translation as Reparation. doi: 10.4324/9781315759777

1.2 Overview of Key Components

For the purposes of this paper, the proposed system is divided into three key components. Firstly, there is the web-framework and user interface through which users and volunteers will interact with the system. Secondly, there is the active learning component. Thirdly, there is the neural machine translation model itself. The full system attempts to draw insights from current research in all three areas to create a novel system prototype.

The web-framework is the component of the system through which users and volunteers propose new languages to be translated, contribute to ongoing data collection initiative by responding to translation prompts and providing translations, and access online translators trained from the current translated data corpus. In addition to the design of the user interface, this component also encompasses the design of the database that is used to store and access what the web-framework knows and learns.

The active learning component of the system determines which sentences the volunteer translators are prompted with. Translation labor is a very expensive and scarce resource,[4] and as the system will be generating the bulk of its data from scratch, we seek to take advantage of the fact that we can choose exactly what sentences are to be translated to make the most out of each sentence. This component will analyze available data (such as the current translated corpus or bodies of monolingual, untranslated text) in order to guess which untranslated sentences might most improve the system if they were known.

The machine translation component of the system is comprised of a neural network that continuously learns from the growing corpus of translated sentences. This component

[4] Bird, Steven. (2019-2020) *Mobile Software for Oral Language Learning in Arnhem Land.* ILA Project Research Proposal.

will be optimized for low-resource data conditions so that the real-time translator provided to communities is effective as soon as possible. Once the data corpus grows large enough to exceed resource-scarcity, a new, more standard neural machine translation network can be trained on the collected data.

1.3 The System in Context: History and Related Work

The problems of crowd-sourcing translation data, active learning in machine translation, and optimizing neural machine translation have been approached before from many different angles and in many distinct settings. A brief overview of the history and related work of these tasks will be given below, as well as an explanation of how this previous work will inform the design and implementation of my system.

1.3.1 Crowdsourcing Translation Data

Obtaining accurate translation data from crowds of bilingual speakers is no trivial task. As Ambati et al. elucidate in their paper "Collaborative Workflow for Crowdsourcing Translation," translating a sentence from one language to another is not an exact science.[5] Often, there may be no exact translation but rather multiple semi-correct translations. Thus, translation accuracy is better represented as a spectrum of accuracy rather than by straightforward 'wrong' and 'right' labels. With this insight in mind, my system will utilize independent translators to cross-reference and correct data as well as to refine or complicate the translation of sentences. Though translation labor is scarce, it is still important to leverage consensus in compiling a solid translated data corpus.

[5] Ambati. "Collaborative Workflow for Crowdsourcing Translation" DOI: 10.1145. Retrieved from https://dl.acm.org/doi/10.1145/2145204.2145382

It is also important to consider the unique challenges of crowd-sourcing in indigenous spaces. The very act of conducting data collection initiatives in these spaces poses certain social and ethical risks if they are not conducted thoughtfully. Initiatives may run the risk of disenfranchising indigenous language authorities or commodifying knowledge and data. Furthermore, forcing language through the process of translation into another can change and even reduce that language and its culture. This is especially true when there is a significant disconnect between the language and the systems being externally imposed upon it, such as written scripts,[6] or significant power disparities between two languages.

Beyond ethical issues, we may also confront several major logistical obstacles in the collection of accurate and consistent data. For instance, literacy rates amongst indigenous communities can be very low. Furthermore, indigenous language is often far more orally-based and far less rigidly defined with strict and universally accepted grammatical conventions, which makes ensuring and validating translation accuracy more complicated.[7] With all this in mind, I will seek to ensure that my system is robust to errors and ambiguity by employing a neural machine translation model that can learn from and weigh many different examples to synthesize accurate translation.

There are also valuable lessons to learn from Google's Translate's crowdsourcing platform: Google Community. In particular, I hope to incorporate their gamified incentive system (which motivates contributors with points and badges) and their diverse

---

[6] Personal conversations with Dr. Steven Bird and Dr. Rachel Nordlinger, director of the University of Melbourne's Research Unit for Indigenous Language

[7] Bird, S. (2019). *Sparse Transcription: Rethinking Oral Language Processing*. Northern Institute, Charles Darwin University.

input scheme that allows contributors to either provide translations to prompts, correct other translations, or directly upload already translated parallel text files.

1.3.2 Active Learning for Machine Translation

In high-resource scenarios, active learning has been demonstrated to greatly refine neural machine translation systems. Some approach this problem as a machine learning classification problem. Peris et al. frames the active learning problem as follows: given "an unbounded stream of source sentences" determine which sentences are worth translating by a human agent and which are not.[8] However, as Liu et al. explain in their paper "Learning to Actively Learn Neural Machine Translation," the traditional active learning heuristics for machine translation are significantly limited when there is very little initial bitext.[9] That being said, effective active learning strategies at the initial, low-resource stages are of the upmost importance, as neural machine translation quality "degrades severely in such settings."[10] One potential avenue to approach this problem is presented by Sennrich et al., who demonstrate that when the amount of bilingual text is limited, leveraging monolingual text can greatly improve the effectiveness of active learning.[11] I will experiment with leveraging monolingual text to improve translation prompts.

---

[8] Peris, Á., & Casacuberta, F. (n.d.). Active Learning for Interactive Neural Machine Translation of Data Streams. https://www.aclweb.org/anthology/K18-1015.
[9] Liu, M., Buntine, W., & Haffari, G. (n.d.). Learning to Actively Learn Neural Machine Translation. Retrieved from https://www.aclweb.org/anthology/K18-1033.
[10] Ibid.
[11] Sennrich, R., Haddow, B., & Birch, A. (n.d.). Neural Machine Translation of Rare Words with Subword Units. https://www.aclweb.org/anthology/P16-1162/.

1.3.3 Machine Translation in Low-Resource Contexts

The task of machine translation has been studied extensively and the obstacle of resource scarcity has been repeatedly confronted. In the past couple years, the attention-based sequence-to-sequence recurrent neural network model has been shown to significantly surpass other machine translation methods, even in fairly low-resource conditions.[12] Some systems have attempted to compensate for resource scarcity by combining statistical or neural machine translation methods with a rule-based approach.[13] This is difficult to generalize to new or unknown languages, however, because rule-based translation systems typically need skilled computational linguists to construct and maintain.[14]

In recent years the supremacy of neural machine translation has been clearly demonstrated and established, so much so that in 2016, Google Translate shifted their system away from statistical machine translation to neural machine translation. Despite the success of recent neural machine translation systems, little scholarship has sought to apply these systems to indigenous languages. The primary reason for this is that neural machine translation systems are often outperformed by other machine translation systems (such as phrase-based statistical machine translation) in very low-resource settings.[15] In their paper "Building Machine Translation Systems for Indigenous Languages," Llitjós et

---

12 Kang, M. VaLaR NMT: Vastly Lacking Resources Neural Machine Translation. *Stanford University*. http://web.stanford.edu/class/cs224n/reports/custom/15811193.pdf

13 Ahsan, A. (n.d.). Coupling Statistical Machine Translation with Rule-based Transfer and Generation. Retrieved from http://www.mt-archive.info/10/AMTA-2010-Ahsan.pdf

14 Font Llitjós, A., & Aranovich, R. Building Machine translation systems for indigenous languages . *Language Technologies Institute: Carnegie Mellon University* . http://www.cs.cmu.edu/~aria/Papers/FontAranovich_CILLA2_mapuche_quechua(2).pdf

15 Sennrich, Rico, Zhang, & Biao. (2019, May 28). Revisiting Low-Resource Neural Machine Translation: A Case Study. https://arxiv.org/abs/1905.11901.

al. detail a project called AVENUE that sought to employ a combination of rule-based and statistical machine translation to two South American indigenous languages: Quechua in Peru and Mapuche in Chile.[16] While the project had many successes, it also required a significant amount of specialized attention, time, and labor from developers and linguists that cannot be afforded to many indigenous languages.

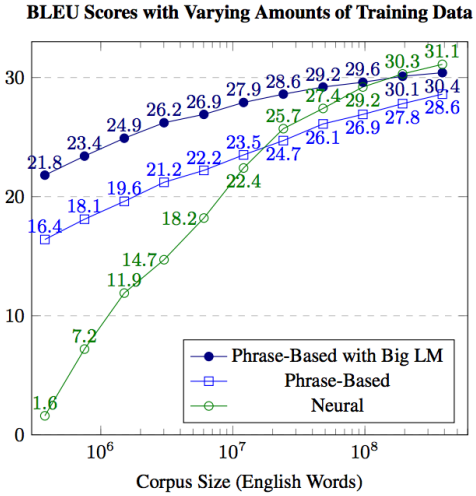
However, recent developments suggest that neural machine translation may have a place in low-resource language translation after all. In their paper " Revisiting Low-Resource Neural Machine Translation: A Case Study," Sennrich et al. demonstrate that neural machine translation systems can outperform statistical machine translation systems with much less data than was previously thought. On the following page, we observe to the right the performance of neural machine translation systems compared to statistical machine translation systems as established by Koehn and Knowles in 2017.[17] To the left, however, we see the performance of neural machine translation (optimized for low-resource conditions by Sennrich and Zhang in 2019) compared to the same statistical machine translation approach.[18]

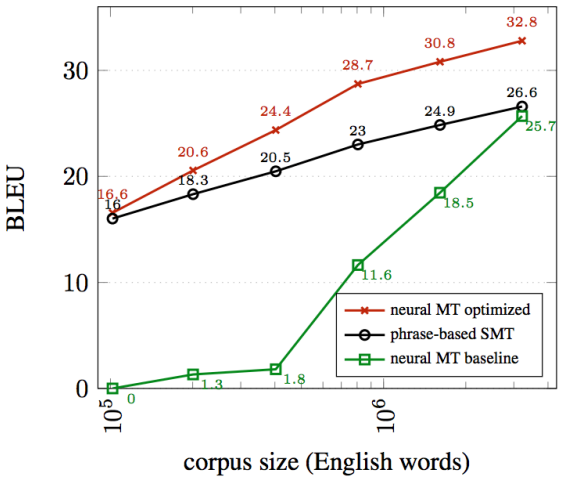
---

[16] Font Llitjós, A., & Aranovich, R. Building Machine translation systems for indigenous languages . *Language Technologies Institute: Carnegie Mellon University* . http://www.cs.cmu.edu/~aria/Papers/FontAranovich_CILLA2_mapuche_quechua(2).pdf

[17] Philipp Koehn and Rebecca Knowles. 2017. *Six Challenges for Neural Machine Translation. In Proceedings of the First Workshop on Neural Machine Translation,* pages 28–39, Vancouver.

[18] Sennrich, Rico, Zhang, & Biao. (2019, May 28). Revisiting Low-Resource Neural Machine Translation: A Case Study. https://arxiv.org/abs/1905.11901.





We observe that before optimizing for specialized low-data conditions, neural machine translation (NMT) systems require over 100 million words (over 5 million sentences) to outperform phrase-based statistical machine translation. However, with optimizations, Sennrich and Zhang demonstrate that "neural machine translation outperforms phrase-based statistical machine translation with as little as 100,000 words of parallel training data."[19] With these insights and results in mind, I chose to design the machine translation system for my web framework as a low-resource optimized neural machine translation system with attention.

2 Approach

In this section, I outline the approaches and design choices I selected for each of the three key components. I also address the data I chose to work with.

[19] Sennrich, Rico, Zhang, & Biao. (2019, May 28). Revisiting Low-Resource Neural Machine Translation: A Case Study. https://arxiv.org/abs/1905.11901.

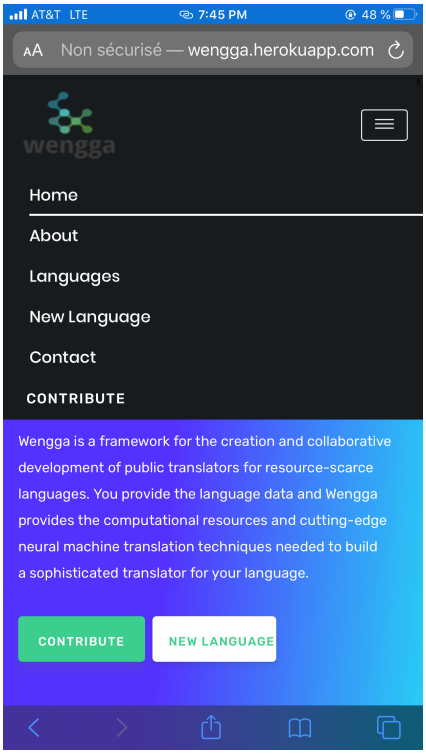
2.1 The Data

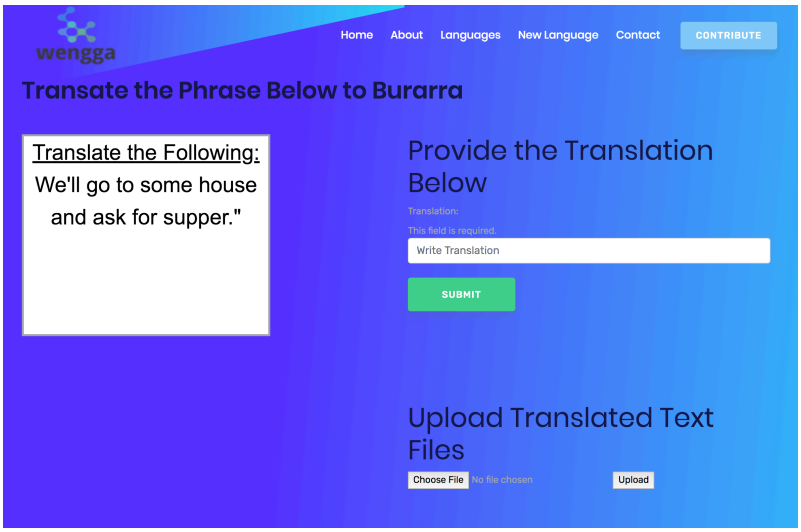
For the purposes of testing and development of the active learning and neural machine translation components of this project, I chose to simulate resource-scarce language by limiting the amount of text data that I use to inform the system. As I speak Spanish and there is an abundance of Spanish to English translated datasets to work with, I chose it as the simulated resource-scarce language. By working with and training on a language that isn't truly resource-scarce, I can continue to add and evaluate text to study the impacts of each of these and observe how much is needed to produce a useful system. Furthermore, for the active learning component, I can utilize very accurate translators (namely, Google Translate) to roughly simulate user annotation and translation of sentences for testing and experimentation.

While generating functional and usable online translators for communities is a major goal of this work, an important component that is valuable in and of itself is the consolidation, parsing, and formatting of indigenous language data. Thus, I also decided to begin collecting and consolidating some resource-scarce data for different Australian indigenous languages that I hope to one day work with by scraping from online dictionaries and already-translated text sources such as the Bible.

2.2 The Web Framework

There was a one principle intention behind most of my design decisions for the web framework: accessibility. For maximum accessibility, I designed a web framework that could be effectively accessed through both desktop computers and mobile devices through a simple and straightforward user interface, as illustrated below.





We observe above several key design and functionality features, such as the ability to 'Contribute' to an existing language or propose a 'New Language' for which to begin amassing data and refining a translator. Toward the bottom of the rightmost image above, we see the option for users to upload already translated text files which, if formatted properly, can be automatically read into the system's database and used to improve translations.

As I dove deeper into the problems confronting indigenous communities and engaged in conversations with Professor Steven Bird, one of the world's leading computational linguists and expert in Australian indigenous language, I gained insights that complicated my definition of accessibility. He explained that indigenous language speakers are not the only, or perhaps even most important, target audience for these translators. Rather than a tool aimed primarily at helping indigenous people integrate into dominant society and

culture, these translators may more effectively serve those outside of indigenous communities to help them better understand, serve, and take part in indigenous communities. For instance, volunteers with the Peace Corps often have the need to learn and study indigenous language before entering communities, and the existence of sophisticated translators would greatly aid their work. With this in mind, I expanded the audience that the website was targeting (formerly just indigenous language speakers) to also include and serve those outside indigenous spaces looking to enter them.

2.3 Active Learning Approaches

The goal of the system's active learning component is to determine the sentences that human translators should be prompted with. In other words: what new data would most improve our NMT model? We consider two different approaches: a frequency-based approach and a perplexity-based approach.

The frequency-based approach analyzes a large corpus of English text to learn the distribution of words and phrases. It then cross-references this knowledge with the corpus of translated text to figure out which of the highest priority (most common) words and phrases are not sufficiently represented in our translated corpus. The intuition here is that this approach helps the model acquire the most important (by frequency) words and phrases as quickly as possible to best improve its applicable coverage of a language. One major limitation here is that we only consider the frequency of words and phrases in English, rather than also analyzing data from the minority language. However, one major benefit of the this approach is that we can deliberately select the large corpus of untranslated text to target specific domains. For instance, by specifically analyzing legal,

medical, or educational texts, we could build a translator that works better in those domains than a general purpose translator.

The perplexity-based approach, on the other hand, begins by analyzing the English corpus that is already translated to construct a language model. It then evaluates the perplexity[20] of new sentences with respect to this language model. The intuition here is that the sentences with the highest perplexity according to the learned language model will ”surprise” our translator the most, and thus they would be the most important sentences to translate.

2.4 Resource Scarce Optimizations

In their paper "Revisiting Low-Resource Neural Machine Translation: A Case Study,” Sennrich et al. establish and describe a handful of optimizations that can be made to neural machine translation systems to help them perform better in low-resource contexts.[21] For instance, they observe that “while the trend in high-resource settings is towards using larger and deeper models, Nguyen and Chiang (2018) use smaller and fewer layers for smaller datasets.” In a similar vein, while previous work has argued for larger batch sizes in NMT” Sennrich et al. find that “using smaller batches is beneficial in low-resource settings.” I chose to experiment with three optimizations to understand how the performance of the network could be improved for low-resource situations by: 1) varying the batch size during training, 2) testing smaller, shallower models, and 3)

[20] Perplexity is a measurement of how well a probability model predicts a sample.
[21] Sennrich, Rico, Zhang, & Biao. (2019, May 28). Revisiting Low-Resource Neural Machine Translation: A Case Study. https://arxiv.org/abs/1905.11901.

complicating language representation such that words below a certain frequency are split into smaller units.

3 Implementation

3.1 Data Collection and Processing

For simulating resource-scarce language data, I use a large database[22] of English-Spanish sentence pairs. For the large corpus of untranslated English text, I use a collection of English novels. This data is all read in through a Python script and preprocessed into the appropriate ASCII encoding, then tokenized using Python's Natural Language Toolkit (NLTK)[23] and regular expression libraries. Finally, to begin collecting indigenous language data, I wrote a Python-based web scraper using Selinium to collect translated words and sentences from an online database created and maintained by Australian Society for Indigenous Languages.[24]

3.2 Implementation of the User Interface

For the prototype website, I used the Python-based web development framework Django.[25] The frontend user interface is written with HTML and JavaScript, using a template from ColorLib as a base,[26] and the database is SQLite within the Django framework. The active learning and neural machine translation components are integrated

[22] Tab-delimited Bilingual Sentence Pairs: Selected sentence pairs from the Tatoeba Project. (n.d.). Retrieved from http://www.manythings.org/anki/.
[23] Bird, S. Python's Natural Language Toolkit (NLTK).
[24] Australian Indigenous Language Online Dictionaries. (n.d.). Retrieved from http://ausil.org.au/node/3717.
[25] Django (Version 1.5) [Computer Software]. (2013). https://djangoproject.com.
[26] Silkalns, Aigars. (2020, January 2). ColorLib HTML Templates.

into the backend an the website is hosted on Heroku servers and can be accessed at https://www.wengga.org/.

3.3 The Active Learning System

Both the frequency-based and perplexity-based approaches to active learning are implemented in Python. The frequency-based method aims to ensure that our model is sufficiently exposed to the most frequent phrases as quickly as possible, while the perplexity-method evaluates possible sentences to see how perplexing they would be to a language model generated from the corpus of all currently translated sentences.

For the frequency-based approach, we first scan through a large corpus of untranslated English text to determine high frequency words and phrases in the English language. This is implemented such that this scan seeks to understand unigram frequencies, bigram frequencies, trigram frequencies, or a combination of the three depending on an input parameter. We next scan through the English corpus that we know is already translated (similarly learning the frequencies of either unigrams, bigrams, trigrams, depending on the settings) to learn the frequencies with which words and phrases are represented in the translated corpus. These frequencies are all mapped into a hash table such that a phrase's frequency and presence in the translated text corpus and English in general (approximated from the large monolingual corpus) can be looked up in constant time. Using this, we can break a candidate sentence down into its words and phrases to give it a score for how useful it would be to the model. If it is above a certain threshold (contains enough words or phrases that are above the high-frequency bar), we select it as a sentence that should be translated.

For the perplexity-based approach, we first scan through the English text whose translations are known in order to train a language model with add-alpha smoothing. We next select a chunk of English text for which no translations are known. For each sentence in this chunk of text, we determine the perplexity according to the language model we learned. If the perplexity is above a certain threshold, we then select this sentence as one that is worth being translated by a human volunteer.

3.4 The Neural Network Architecture

For the neural machine translation network, I built a Tensor-flow-based sequence-to-sequence neural network with Bahdanau attention and a sparse softmax cross entropy loss function by following an online tutorial provided by TensorFlow.[27] After building and testing the base network I began to experiment with several adaptations and optimizations to increase the performance in very low-resource settings, drawing from the insights from "Revisiting Low-Resource Neural Machine Translation: A Case Study." First, I experimented by varying parameters such as the batch size during training. I systematically modified batch size, retrained the network, and evaluated the resultant networks' performances using a metric I will describe in the evaluation section of this paper. I also experimented with training shallower models with fewer layers, retraining and evaluating as above. Finally, I implemented a basic subword language representation, following the intuition that "in low-resource settings, large vocabularies result in low-frequency (sub)words being represented as atomic units at training time, and the ability to learn good high-dimensional representations of these is doubtful. Sennrich et al. (2017) propose a minimum frequency threshold for subword units, and splitting any less

[27] Neural machine translation with attention: TensorFlow Core. Retrieved from https://www.tensorflow.org/tutorials/text/nmt_with_attention.

frequent subword into smaller units or characters." Following this, I scanned the training corpus to learn the frequencies of all words. For all words with a frequency less than two (a threshold that seemed to perform best during experimentation), I split these words into smaller chunks of five characters.

4 Evaluation

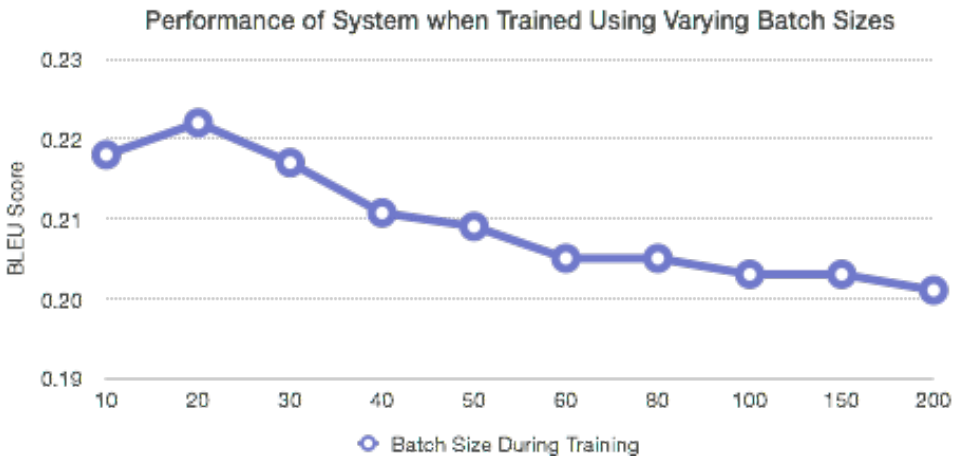
4.1 Evaluating the Neural Machine Translation Component

To initially evaluate the neural machine translation system during development, I performed a series of manual tests to see if the model successfully translated a series of Spanish sentences to English. With a small training sample of only 30,000 sentences, the model was able to effectively translate many simple test sentences, albeit with obvious errors.

However, to more rigorously evaluate the model and understand how its performance is affected by different modifications, I wrote a Python script that evaluates a model's accuracy by having it translate a validation corpus: a set of sentences for which true translations are known. The metric I selected is the BLEU (bilingual evaluation understudy) score, which is a standard machine translation evaluation metric. I then use the ground truth translation to compute the smoothed BLEU scores for each of these translations (giving equal weight to unigrams, bigrams, trigrams, and tetra-grams) and take the average of these over the testing corpus to represent the models accuracy. For each of the following experiments, I used a training corpus of 30,000 sentences and a validation corpus of 5,000 sentences. For the baseline performance evaluation of the model, I used a model of 500 layers, set the batch size to 60, and used no subword

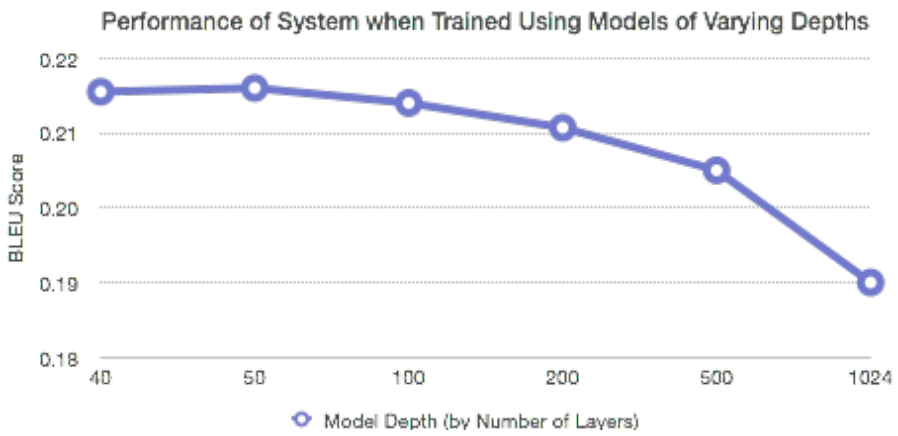
representation. The average performance of this baseline model resulted in a BLEU score of .205.

The first model modification I explored was changing the batch size during training. Keeping the number of layers fixed at 500, I began varying the batch size before retraining and re-valuating the model.



As can be seen in the graph above, the results I got confirmed the theory presented by Sennrich et al.: in low-resource settings, very small batch sizes seem to perform better than larger ones. In this training scenario, the optimal batch size was 20, which performed over 6% better than the baseline.

I next explored the effects of model depth on performance in low-resource settings. Keeping the batch size fixed at 60, I trained and evaluated models of varying depth to see which performed best.



Again, the results I got confirmed the theory presented by Sennrich et al.: in low-resource settings, smaller models seem to perform better than larger ones. In this training scenario, the optimal model depth seemed to be around 50 layers, which performed over 5% better than the baseline.

Finally, I evaluated the subword representation in which infrequent words were split into smaller chunks. With this language representation, a model of 500 layers trained using a batch size of 60 actually performed worse than the baseline, falling from a BLEU score of .205 to score of .188: over an 8% decrease in performance. I believe this is because my subword representation was more simplistic than the one referenced by Sennrich et al.

4.2 Evaluating the Active Learning Component

To evaluate the two approaches to active learning, I chose to implement an intrinsic evaluation scheme in order to allow us to study the active learning component's performance without assuming any sort of machine translation system (such as statistically or neurally based machine translation system, for instance). For each

heuristic, I designed an intrinsic metric to gauge how effectively the approach would be at preparing the system to translate an unseen corpus.

For the frequency-based approach, I designed an evaluation metric I call coverage in order to evaluate a system's coverage of the most frequent words and phrases on an unseen text corpus. To calculate the coverage of a system, I iterate through all unigrams, bigrams, and trigrams of the unseen text to determine what percentage of them are 'covered.' A word or phrase is considered covered if the model has seen, during training, two or more instances of it. I then trained two models– one trained on 30,000 sentences that are actively learned using the frequency-based approach and another trained on 30,000 sentences chosen randomly– and compared their coverage on an unseen set of 10,000 sentences. The results: the system trained with the frequency-based approach had a 36% coverage of the unseen text while the randomly trained system had only a 22% coverage.

I next evaluated the perplexity-based approach by comparing the perplexity of an unseen text corpus with respect to a language model constructed from an actively learned training corpus versus a language model constructed from a randomly learned training corpus. The intuition here is that whichever model is the least perplexed by the unseen corpus is superior. Again, I trained two models– one with 30,000 sentences that were actively learned using the perplexity-based approach and another on 30,000 sentences chosen randomly– and then compared the perplexity of each with respect to an unseen set of 10,000 sentences. The results: the system trained with the perplexity-based approach was 9% less perplexed by

the unseen sentences than the randomly trained system, meaning that it would be better able to predict that corpus.

4.3 Plan to Evaluate the User-Interface

Though I did not have the opportunity to conduct beta-testing this past semester to evaluate the user interface, I hope to lead a beta-test next semester in Northern India. I plan to travel to the small city of Panipat, India and work with the faculty and students of the DAV Thermal Colony public school. The members of this community speak both English and Haryanvi, a language of the Western Hindi group that is currently not supported by machine translators. The Principal of the school, Ms. Ritu Dilbagi, is excited to support my research and schedule time for her students and faculty to contribute to the system. She would also connect me with adults and children willing to report on their experiences with Haryanvi and the different ways they could be supported with language tools.

I also hope to receive a grant to further develop and launch this system in Northern Australia under the guidance of Dr. Steven Bird. He is currently leading indigenous language data collection and translation field work in this area, and I am excited to learn how he faces the challenges of working and researching in this space. The image below shows a snapshot of a setup he designed to collect language data from a rural indigenous community with very limited access to internet.

5 Conclusions

5.1 Reflections

This prototype system has achieved several key accomplishments. Firstly, the web framework is functional and ready for beta-testing on desktop computers or mobile devices. Secondly, as has been demonstrated, the active learning component prompts users to generate a translated text corpus with important key properties such as coverage of key words and phrases. Finally, the neural machine translation component has been adapted and shown to perform more effective translation in low-resource conditions.

Despite these results, there are still significant concerns and complications to consider when it comes to truly launching this system. For instance, one potential concern is that the languages used for development and testing (English and Spanish) are both Western European languages that are more closely related than an indigenous language is likely to be to a Latin or Germanic language.

5.2 Implications and Impact

This project is a prototype for an initiative I hope to truly launch one day in Northern Australia. The first language I am planning to target and develop a machine translator for is Australian Kriol, but I hope to one day expand this to other indigenous languages. I envision this work having four concrete impacts. First, from a computational research standpoint, the development of thoughtful, new approaches to indigenous language data collection and more sophisticated models for understanding the available data will help to redefine how computational

linguists approach this task. Secondly, from a linguistic standpoint, the consolidation and structuring of indigenous language data is an incredibly important endeavor. With the rate at which knowledge of these languages is diminishing, it is vital that these platforms and initiatives begin as quickly as possible. Thirdly, from an anthropological standpoint, this research would serve scholars who do not speak indigenous languages by enabling them to overcome language barriers and connect with the rich history of indigenous people and indigenous people themselves. Finally, from a social standpoint, this work will have concrete consequences on the lives and on the social mobility of indigenous language speakers. Access to online translators and language education platforms will be a huge resource to empower individuals to more easily navigate society and take advantage of educational resources online and beyond. It would also serve as a priceless tool for language learning and would aid communities as they work to educate children and keep their language alive.

5.3 Future Work

I hope that this prototype will inform the system I hope to one day launch to begin effective data collection and machine translation for indigenous and resource-scarce language. Some major milestone goals I have set as I work toward that point include advancing this research to develop even more sophisticated and effective active learning and low-resource machine translation approaches. I hope to develop a more complex and successful active learning system as part of my final project for Princeton's Natural Language Processing course that builds on this paper's approaches, and I would one day like to develop a machine translation model that

can share information between indigenous languages of the same family. I would also like to deploy this application for iOS and Android systems (with the gamified ranking and award system mentioned earlier) so that they are more accessible and can be more easily and consistently used.

6 Acknowledgements

I am so grateful to Professor Srinivas Bangalore for advising this project and counseling me as I attempt to chart a path that allows me to put this research into action! His enthusiasm for my enthusiasm and for this work has helped me to believe that I could really make an impact one day with my research and studies, and I hope one day I will be able to do so and make him proud! I am also very grateful for all the insight that Dr. Steven Bird and Dr. Rachel Nordlinger shared with regard to machine translation and indigenous language in Australia. Their expertise and wisdom was invaluable!

Thank you!

7 Bibliography

Ahsan, A. (n.d.). Coupling Statistical Machine Translation with Rule-based Transfer and Generation. Retrieved from http://www.mt-archive.info/10/AMTA-2010-Ahsan.pdf

Australian Indigenous Language Online Dictionaries. (n.d.). Retrieved from http://ausil.org.au/node/3717.

Tab-delimited Bilingual Sentence Pairs These are selected sentence pairs from the Tatoeba Project. (n.d.). Retrieved from http://www.manythings.org/anki/.

Bandia, P. (2014). Translation as Reparation. doi: 10.4324/9781315759777

Bird, S. (2019). *Entering the Life-World of an Indigenous Community: An Autobiographical Design Journey.* Northern Institute, Charles Darwin University. In-Progress Draft Shared Privately.

Bird, Steven. (2018) *Learning English and Aboriginal Languages for Work. Research Proposal* Submitted to ARC.

Bird, Steven. (2019-2020) *Mobile Software for Oral Language Learning in Arnhem Land.* ILA Project Research Proposal.

Bird, S. (2019). *Sparse Transcription: Rethinking Oral Language Processing*. Northern Institute, Charles Darwin University.

Font Llitjós, A., & Aranovich, R. Building Machine translation systems for indigenous languages . *Language Technologies Institute: Carnegie Mellon University* . Retrieved from: http://www.cs.cmu.edu/~aria/Papers/FontAranovich_CILLA2_mapuche_quechua(2).pdf

Gavrila, M., & Vertan, Training Data in Statistical Machine Translation - the More, the Better? Retrieved from https://www.aclweb.org/anthology/R11-1077.

Kang, M. (n.d.). *VaLaR NMT: Vastly Lacking Resources Neural Machine Translation.* Stanford University. http://web.stanford.edu/class/cs224n/reports/custom/15811193.pdf

Liu, M., Buntine, W., & Haffari, G. (n.d.). Learning to Actively Learn Neural Machine Translation. Retrieved from https://www.aclweb.org/anthology/K18-1033.

Neural machine translation with attention : TensorFlow Core. (n.d.). Retrieved from https://www.tensorflow.org/tutorials/text/nmt_with_attention.

Philipp Koehn and Rebecca Knowles. 2017. *Six Challenges for Neural Machine Translation. In Proceedings of the First Workshop on Neural Machine Translation,* pages 28–39, Vancouver.

Peris, Á., & Casacuberta, F. (n.d.). Active Learning for Interactive Neural Machine Translation of Data Streams. Retrieved from https://www.aclweb.org/anthology/K18-1015.

Sennrich, R., Haddow, B., & Birch, A. (n.d.). Neural Machine Translation of Rare Words with Subword Units. Retrieved from https://www.aclweb.org/anthology/P16-1162/.

Sennrich, Rico, Zhang, & Biao. (2019, May 28). Revisiting Low-Resource Neural Machine Translation: A Case Study. Retrieved from https://arxiv.org/abs/1905.11901.

Silkalns, Aigars. (2020, January 2). ColorLib HTML Templates.

Tab-delimited Bilingual Sentence Pairs: Selected sentence pairs from the Tatoeba Project. (n.d.). Retrieved from http://www.manythings.org/anki/.

Ward, M. (n.d.). *CALL for Endangered Languages: Challenges and Rewards.* Retrieved from:https://www.researchgate.net/publication/248906565_CALL_for_Endangered_Languages_Challenges_and_Rewards.