Cross-Lingual Wolastoqey-English Definition Modelling

Diego Bear
Faculty of Computer Science
University of New Brunswick
diego.bear@unb.ca

Paul Cook
Faculty of Computer Science
University of New Brunswick
paul.cook@unb.ca

Abstract

Definition modelling is the task of automatically generating a dictionary-style definition given a target word. In this paper, we consider cross-lingual definition generation. Specifically, we generate English definitions for Wolastoqey (Malecite-Passamaquoddy) words. Wolastoqey is an endangered, low-resource polysynthetic language. We hypothesize that sub-word representations based on byte pair encoding (Sennrich et al., 2016) can be leveraged to represent morphologically-complex Wolastoqey words and overcome the challenge of not having large corpora available for training. Our experimental results demonstrate that this approach outperforms baseline methods in terms of BLEU score.

1 Introduction

Definition modelling, introduced by Noraset et al. (2017), is the task of automatically generating a dictionary-style definition for a given target word. Definition modelling can provide more-transparent, human-interpretable representations of the information in embeddings. Definition modelling could also potentially be applied to automate, or semi-automate, the constructing or updating of dictionaries, for example, by generating draft definitions for newly-emerged words that are not yet listed. Although there has been a range of work on definition modelling (e.g., Ni and Wang, 2017; Gadetsky et al., 2018; Chang and Chen, 2019) the focus has been on monolingual definition modelling, with the target word and generated definition being in the same language.

Malecite-Passamaquoddy (also Maliseet-Passamaquoddy, Passamaquoddy-Maliseet) is an Eastern Algonquian language spoken in regions of what is now New Brunswick and Quebec, Canada, and Maine, United States. Malecite and Passamaquoddy are dialects of this language. However, *Malecite* is a Mi'kmaq exonym, with *Wolastoqey* being the term this speech community uses to refer to their language. We therefore use the term *Wolastoqey* throughout this paper.

Wolastoqey is an endangered language, with roughly 300 remaining first language speakers in Canada (Statistics Canada, 2017). Moreover, children are typically not learning the language proficiently. Wolastoqey is also a low-resource language, with no large corpora or annotated datasets available for training natural language processing (NLP) systems. However, the Passamaquoddy-Maliseet Dictionary (Francis and Leavitt, 2008) is available online through the Passamaquoddy-Maliseet Language Portal.[1] This dictionary includes roughly 19k entries with Wolastoqey headwords and English definitions. Many entries also include parallel Wolastoqey-English example sentences. There has been very little prior computational work on Wolastoqey, with Farber (2015) presenting a preliminary finite-state model of nouns.

Wolastoqey, like other Algonquian languages, is polysynthetic. Verbs in particular have rich morphological structure, and often include several roots (Leavitt, 1996). Consider the example gloss below for *paskoloqessu*:

pask-oloq-ess-u
breaking-ice-move.quickly-s/he
'She or he moves quickly across ice as it cracks'

The root *oloq* can be seen in various other words, such as *'ketoloqtehmon* 's/he chips it out of ice', *sahsoloqe* 'it is slippery, is icy', and *supoloqe* 'there is smooth ice (on lake, etc.)'. There is, however, ambiguity in that the character sequence *oloq* does not always correspond to this morpheme. For example, in *oloqapeku* 's/he crawls in that direction' *oloq* has the meaning of 'in that direction'.

[1] Passamaquoddy-Maliseet Language Portal (http://www.pmportal.org); Language Keepers and Passamaquoddy-Maliseet Dictionary Project.

Proceedings of Recent Advances in Natural Language Processing, pages 138–146
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_017

All examples are taken from the Passamaquoddy-Maliseet Language Portal.

In this paper we propose a model for cross-lingual Wolastoqey-English definition modelling. We hypothesize that sub-word representations of Wolastoqey words based on byte pair encoding (BPE) tokenization (Sennrich et al., 2016) can be leveraged to generate English definitions. We propose a sequence-to-sequence model (Sutskever et al., 2014) in which the encoder operates over Wolastoqey words segmented via BPE, and the decoder generates English definitions. We show that our proposed model is able to outperform baseline systems in terms of BLEU score.

Wolastoqey speakers regularly create new words by creatively combining roots (Leavitt, 1996). As such, not all words can be expected to be included in a dictionary. Cross-lingual Wolastoqey-English definition modelling could therefore be helpful for Wolastoqey learners.

2 Related Work

The task of definition modelling is to learn to generate a dictionary-style definition for a given input word. This task was initially described by Noraset et al. (2017), who focused on generating English definitions for English words. Noraset et al. proposed a word-to-sequence neural language model, composed of a two-layer LSTM and a parallel CNN, to generate a definition given an initial input word and its embedding. This language model-based approach generates definitions by iteratively predicting the next occurring word given some prior history. In this model, different inputs were given to each of the sub-components. The LSTM component was initially given the embedding for the word being defined, but also considered the embeddings of its previous output in the form of context at a given timestep. The CNN sub-network, on the other hand, was used to extract character-level information about the word and, as such, was given the characters of the word being defined. The CNN was included because the word-level LSTM has no knowledge of sub-word information. Manual analysis of the proposed system's performance considered seven types of errors that were observed to occur in the generated definitions. These include redundancy, self-reference, wrong part-of-speech, under-specification, opposite definition, close semantic errors, and incorrect definition generation. Out of all these errors, incorrect definition generation was observed to be the most common. This paper further found that high-quality word embeddings were crucial for definition modelling to be successful.

One challenge for our work is that we do not have a large corpus available from which to learn high-quality Wolastoqey word embeddings. We propose to use BPE segmentation to overcome this. Specifically, we hypothesize that sub-word representations based on BPE can be leveraged to represent morphologically-complex Wolastoqey words without requiring a large corpus to be available for training word-level embeddings.

The approach of Noraset et al. (2017) is context-agnostic; i.e., the model generates a definition for a target word without any specific context of usage for the target. Other context-agnostic approaches to definition modelling include Yang et al. (2020) who incorporate knowledge of Chinese sememes (minimum semantic units) for Chinese definition modelling, and Balachandran et al. (2018) who propose a domain-specific definition generation model for the software domain. In line with these previous studies, we also propose a context-agnostic approach.

Contrasting with context-agnostic approaches, context-aware approaches to definition modelling have also been considered (e.g., Ni and Wang, 2017; Gadetsky et al., 2018; Mickus et al., 2019). In these approaches a definition is generated for a target word used in a specific context. Some context-aware methods have used a sequence-to-sequence model (Ni and Wang, 2017; Mickus et al., 2019) as does our proposed approach. Ni and Wang propose a sequence-to-sequence model to generate definitions for non-standard English words. Their encoder uses a character-level LSTM to represent the target word and a word-level LSTM to represent the context. An LSTM is also used for decoding. Our proposed approach is similar to that of Ni and Wang, but we do not use an LSTM to encode context, and our LSTM which encodes the target word operates over BPE tokenization as opposed to characters.

An alternative line of research considers definition extraction (e.g., Navigli and Velardi, 2010) in which sentences containing terms and their corresponding definitions are automatically identified in corpora. We focus on definition modelling, as opposed to extraction, because there are very few corpora containing English definitions of Wolasto-

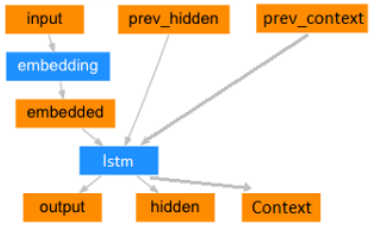

Figure 1: The structure of the encoder.

qey words on which to apply a definition extraction method, and because Wolastoqey is a polysynthetic language and as such many possible words would not be expected to be found in corpora.

3 Model

Cross-lingual definition modelling can be seen as a machine translation task which involves translating a word in a source language to a definition in some target language. We therefore consider using a network architecture proposed for the task of translation rather than a word-to-sequence model proposed in previous work on monolingual definition modelling (e.g., Noraset et al., 2017). Specifically, we consider a sequence-to-sequence model that makes use of an attention decoder (Bahdanau et al., 2014). We base our model on a sample sequence-to-sequence translation model.[2]

3.1 Encoder Architecture

For our encoder model's architecture, we use a simple recurrent neural network consisting of an embedding layer followed by a long-short term memory (LSTM) layer. The structure of the encoder is shown in Figure 1.

The embedding layer of our model serves the purpose of representing the meaning of the subword tokens that compose our vocabulary. To obtain the embeddings used by our encoder, we consider the approach of initializing the weights of our embedding layer to zeroes as well as the approach of first pretraining our embeddings on a corpus of example sentences extracted from the Passamaquoddy-Maliseet Dictionary. While training, we allow the weights of the embeddings to be updated through gradient descent, but also consider freezing these weights in the case of pretrained

[2] https://github.com/spro/practical-pytorch

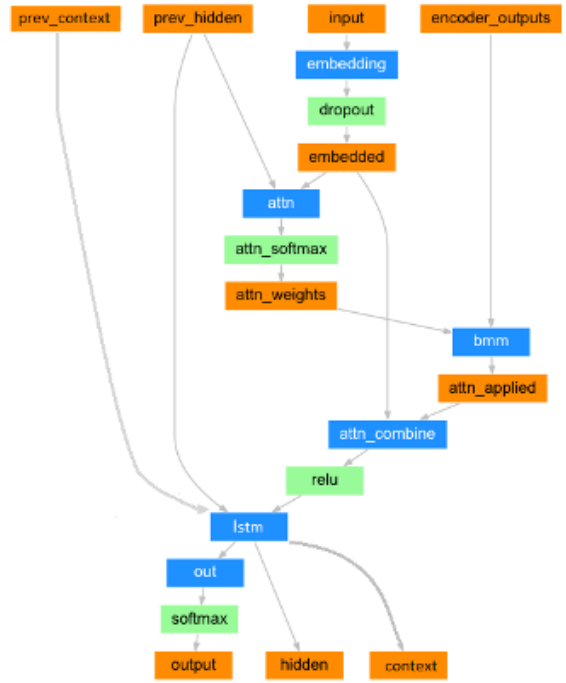

Figure 2: The structure of the decoder.

embeddings. This is done to further analyze the effects pretraining has on system performance.

At a given timestep, input is passed into our encoder in the form of a sequence of indices corresponding to sub-word representations in our input vocabulary. At a given time-step, the encoder will consider a given input subword. For this subword, the encoder will start by looking up its embedding using the embedding layer. This embedding will then be passed to the LSTM layer. From here, the LSTM layer will use this embedding, the hidden layer of the previous time-step and the context from the previous time-step layer to calculate the value for the current hidden layer which acts as the decision at the current timestep. The context is then updated with the information regarding the current decision, and then passed forward with the output at the current time-step. Once all of the outputs have been calculated, we pass the encoder outputs to the decoder.

3.2 Decoder Architecture

The decoder architecture is shown in Figure 2. Rather than relying on a single vector to contain all information about the input sequence, we instead use an attention decoder to consider the encoder outputs more holistically. Our decoder consists of an embedding layer, two intermediary linear layers, a recurrent LSTM layer and a final linear layer

which we then softmax over to get the output at a given timestep. We apply dropout regularization to the embedding layer in an effort to avoid overfitting.

At a given timestep, input is given to our decoder in the form of the decision made at the previous step, or, in the case of the first decision, a start of sequence token, and will take the form of an index of a token contained within our output vocabulary. We then use this index to look up the embedding for the word through the embedding layer. From here, we concatenate the embedding with the hidden state from the previous decision. We pass this value to the first linear layer which will give us weights we can then compare to the encoder outputs through batch matrix multiplication. This product will then be passed to another linear layer with a ReLU activation function which will set all negative values to 0. This value is then passed as input to our LSTM layer, which will also consider the previous hidden layer and the context from the previous time-step. This will give us updated context and hidden states, which we will pass forward to the next timestep. However, to get the current output, we will need to pass the resulting vector from the recurrent component of our system to another linear layer. The results from this layer will then be softmaxed to give us the final output, and our decision for the current timestep.

3.3 Model Variations

In addition to the base model described above, we also consider two architecture variations designed to analyze the effect of model complexity on performance. Specifically, we consider replacing the unidirectional LSTM layers used in both the encoder and decoder with comparatively simpler GRU layers, and comparatively more complex bidirectional LSTM layers. We consider this because we have a relatively small training corpus which might not be large enough to adequately train more-complex models. Although bidirectional LSTMs have been shown to perform better than GRUs and unidirectional LSTMs when sufficient training data is available, they generally perform worse when insufficient training data is available. By considering these model variations, we can attempt to determine whether we have enough training data to justify the use of more-complex models.

4 Experimental Setup

In this section, we describe the dataset constructed for these experiments, the evaluation methodology used, and implementation details for our proposed model.

4.1 Dataset

Wolastoqey is a low-resource language. As such, we are limited in regards to our choice of dataset. For cross-lingual Wolastoqey-English definition modelling we require a dataset consisting of Wolastoqey headwords and their corresponding English definitions. The Passamaquoddy Maliseet Dictionary consists of Wolastoqey head words and their corresponding English definitions. Many entries also include parallel Wolastoqey-English example sentences. We use the headwords and definitions to construct our dataset. We use the example sentences to train embeddings, as well as our BPE tokenizer in the case of Wolastoqey.

The Passamaquoddy-Maliseet Dictionary is available online. There is, however, no publicly available download for the contents of this dictionary. We therefore use Selenium, a web automation tool, to crawl the dictionary and extract the entries.

After scraping the dictionary content, we normalize the text. For this, we remove any entries containing errors such as *#NAME?* as a headword. As each headword can have multiple definitions associated with it, we split definitions on semi-colons as they are used as the primary definition delimiter. We perform a similar operation for the headwords themselves, as an entry can include multiple word-forms for the headword. We split the headword text on commas, with each extracted headword being used to create word-definition pairs with respect to all available definitions for a given word.

Wolastoqey has four parts-of-speech (POS): nouns, pronouns, verbs, and particles (Leavitt, 1996). For this dataset we only include headwords that are nouns, verbs, and particles because there are relatively few entries that are pronouns.

This extraction method produces a dataset that contains $22.5k$ headword-definition pairs from $19k$ valid entries. We then split these pairs into training, development, and test sets. To do this, we first group the data based on headwords to prevent any headword with the same form from appearing in more than one of the sets. Once we have grouped the headwords, we then split the data by headword with 80%, 10%, and 10% of headwords in the train-

ing, development, and test sets, respectively. The training set is used for training our model. The development set is used for model tuning. The test set is held out for final evaluation.

In addition to the dictionary headwords, we also extract $18.6k$ Wolastoqey-English sentence pairs pulled from all valid dictionary entries containing example sentences. As we use these example sentences to train our embeddings, we split this parallel corpus into separate monolingual Wolastoqey and English corpora consisting of $80.5k$ and $181.9k$ tokens, respectively.

4.2 Evaluation

Following previous work on definition modelling (Noraset et al., 2017; Ni and Wang, 2017; Gadetsky et al., 2018) we use BLEU score (Papineni et al., 2002) for evaluation. At test time, we generate an English definition for each Wolastoqey headword (in either the development or test set) and calculate the BLEU score between this generated definition and the gold-standard reference definition for this headword.

We compare our proposed model against a baseline that outputs a randomly selected definition from the development set for any input. This baseline will always produce a syntactically well-formed definition, but the definition is unlikely to be semantically appropriate. We consider two variations of this baseline, POS-aware and POS-agnostic, which differ with respect to knowledge of the POS of the input. The POS-agnostic baseline simply outputs a randomly selected definition. The POS-aware baseline outputs a randomly selected definition corresponding to a headword with the same POS as the input.[3]

We implement our proposed model using Pytorch 1.7.1. We use a hidden layer size of 256 for each layer in both the encoder and decoder sub-models. Our decoder sub-model's dropout layer uses a dropout rate of 0.1. To train our model, we use an Adam optimizer with a learning rate of 5e-4. We train all our models for a total of $500k$ iterations. We use teacher forcing as our training regimen.

To obtain sub-word representations we use the Huggingface Tokenizers 0.10.1 library. For English, we use the pretrained word-piece DistilBERT (Sanh et al., 2019) tokenizer. For Wolastoqey, we train a BPE tokenizer on the Wolastoqey example sentences extracted from the Passamaquoddy-Maliseet Dictionary.

To learn pretrained embeddings we consider both word2vec skip-gram with negative sampling (Mikolov et al., 2013) and fastText (Bojanowski et al., 2017). We use the Gensim version 3.8.3 implementations of skip-gram and fastText. We use the default parameter settings (i.e., a window size of 5 sub-word tokens and 100 dimensional embeddings) except for the minimum frequency to be included in the embedding matrix, which we set to 1 (as opposed to the default of 5) because of the small size corpora we use for training.

We calculate BLEU score using the implementation available in NLTK 3.5 (Bird et al., 2009).

5 Results

In this section we first consider tuning the vocabulary size for the Wolastoqey BPE tokenizer in experiments on development data (Section 5.1). We then present our main results on test data, including an analysis of the impact of pretraining embeddings (Section 5.2) and results for the model variations presented in Section 3.3 (Section 5.3). We then present a qualitative evaluation of the model (Section 5.4).

5.1 Tuning Vocabulary Size

We conduct a grid search to find the optimal vocabulary size for our Wolastoqey BPE tokenizer. The BPE vocabulary will directly affect how our Wolastoqey input words are tokenized and could potentially drastically impact the performance of our proposed model. As we are investigating whether sub-word representations can be leveraged for Wolastoqey-English definition modelling, it is important to determine the optimal vocabulary size.

We consider vocabulary sizes from 2500 to 15000 in increments of 2500. Results on development data are shown in Figure 3. We observe that the optimal vocabulary size is 7500 sub-word tokens. We also observe a steep drop-off in performance when using a vocabulary size that exceeds 7500 tokens. We further observe that the relative performance across parts-of-speech is similar regardless of vocabulary size. We use a vocabulary size of 7500 for the Wolastoqey BPE tokenizer for the remainder of the experiments. At this vocabulary size, each Wolastoqey verb, noun, and particle in the development data is represented by an average of 3.21, 2.56, and 1.93 sub-word tokens,

[3] Although we compare against a POS-aware baseline, the proposed model itself has no knowledge of POS.

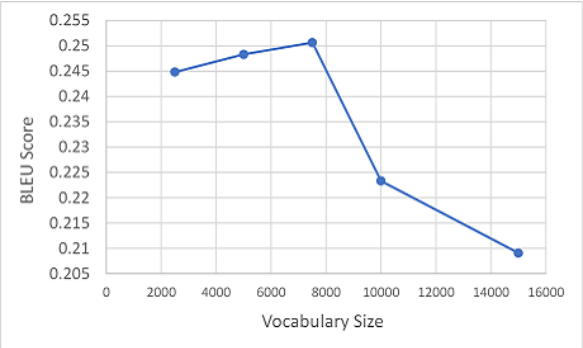

Figure 3: BLEU score for various Wolastoqey BPE vocabulary sizes on development data.

Model	Overall	Verbs	Nouns	Particles
Baselines				
POS Agnostic Baseline	0.144	0.156	0.036	0.042
POS Aware Baseline	0.173	0.184	0.045	0.073
Base Models				
Base Model	0.277	0.304	0.062	0.104
Base Model (verbs only)	0.306	0.333	0.045	0.050
Adjustable Pretrained				
word2vec (Examples)	0.304	0.336	0.080	0.092
word2vec (News)	0.233	0.257	0.064	0.082
fastText (Examples)	0.299	0.327	0.058	0.085
Frozen Pretrained				
word2vec (Examples)	0.148	0.156	0.041	0.070
fastText (Examples)	0.209	0.223	0.045	0.045

Table 1: BLEU score for each model overall and for each POS.

respectively.

5.2 Effects of Pretraining

Results on test data for the proposed model and the baselines are shown in Table 1. We first consider the baselines. The POS-aware baseline, as expected, outperforms the POS-agnostic baseline, overall and for each POS.

The base model (i.e., the model proposed in Sections 3.1 and 3.2) outperforms both baselines. This finding demonstrates that sub-word representations of Wolastoqey words based on BPE can be leveraged to generate English definitions. We note that the performance is much better on verbs, which are the most common POS in our datasets, than other parts-of-speech. We hypothesize that this is because Wolastoqey verbs often consist of multiple morphemes, and indeed are on average split into more sub-word units than other parts-of-speech in the analysis in Section 5.1, whereas other parts-of-speech often correspond to a single morpheme, and are split into fewer sub-word units. We consider training the base model on only verbs, shown as "Base Model (Verbs Only)" in Table 1. Here we see a slight improvement in performance over the base model on verbs, and a corresponding reduction in performance on other parts-of-speech.

The base model does not use pretrained embeddings. We now consider experiments using pretrained embeddings, in which we allow the embeddings to be adjusted through updating during training ("Adjustable Pretrained" in Table 1). We consider word2vec and fastText embeddings trained on the Wolastoqey and English corpora built from the example sentences in the Passamaquoddy-Maliseet Dictionary ("word2vec (Examples)" and "fastText (Examples)", respectively). Both of these approaches improve over the base model in terms of overall BLEU score. For English, because it is a high-resource language, we have access to many sources of embeddings which are pretrained on much larger corpora. We therefore consider using English word2vec embeddings pretrained on text from Google News, shown as "word2vec (News)". This requires switching from word-piece tokenization to word-level tokenization for English. For Wolastoqey, we still use BPE tokenization and train on the corpus of Wolastoqey example sentences. Although these English embeddings are trained on a much larger corpus, this does not yield an improvement over using embeddings pretrained on the English example sentences.

Finally, we consider the impact of allowing the embeddings to be updated during training. We again consider word2vec and fastText trained on the corpora of Wolastoqey and English example sentences, but here we freeze the embedding weights when training the model ("Frozen Pretrained" in Table 1). These methods perform poorly compared to the base model, and compared to the case where the embeddings are updated during training. In particular, here the word2vec embeddings perform roughly on par with the POS agnostic baseline. These findings indicate the importance of allowing the embeddings to be updated during training.

The base model substantially outperforms both baselines considered. In the following subsection we consider further variations on the base model.

5.3 Model Variations

Table 2 shows results on test data for the base model using a (unidirectional) LSTM (i.e., the base model

Model	Overall	Verbs	Nouns	Particles
GRU	0.291	0.322	0.075	0.082
LSTM	0.277	0.304	0.062	0.104
Bidirectional LSTM	0.264	0.285	0.089	0.126

Table 2: BLEU score for each model variant.

presented in Table 1) and GRU, and a bidirectional LSTM. We observe that using a GRU in-place of an LSTM gives a better overall BLEU score. Despite being more powerful, the bidirectional LSTM performs worse overall than the unidirectional LSTM base model. We hypothesize that, because of the relatively small size of the training data, simpler models, such as a GRU, can be more effectively trained. This finding, combined with the findings from Section 5.2, suggests that there could be scope for further improvement through the use of pre-trained embeddings with a GRU.

5.4 Qualitative Analysis

While BLEU score provides a method of empirically evaluating our system, we also wish to perform a qualitative analysis of our system's outputs. For this analysis, we generated definitions for 20 randomly-selected test set Wolastoqey words and manually compared the generated definitions to their ground-truth reference definitions. This analysis was carried out by the first author of this paper, an English first language speaker and Wolastoqey learner. We analyzed the definitions with respect to both semantics and syntax.

For semantics, we considered whether the generated definitions were topically-related to the reference definitions. Of the 20 definitions, 10 were determined to have little to no topical relatedness to the reference definition, 8 showed some level of topical relatedness to their ground truth reference, and 2 were determined to be reasonable definitions for their respective words. For this analysis, we consider reasonable definitions to be definitions that contain few or no syntactic errors and do not significantly vary in meaning when compared to their ground truth references. An example of a word our system is able to reasonably define with little error is *'t-uwapolokehkimal*, which the Passamaquoddy-Maliseet Dictionary defines as 's/he instructs h/ incorrectly; s/he teaches h/ incorrect information, etc.' For this word, our system generates the definition 's/he teaches h/ incorrectly'. An example of a generated definition that shows some level of topical relatedness to the reference definition can be seen for the verb *kcitawse*, for which our system generates the definition 's/he walks without walking, walks in' whereas the reference definition is 's/he walks far into or sinks into, s/he gradually works way into'. In this example, both definitions share some reference to the action of walking; however, the meaning of the generated definition deviates from its ground truth reference.

Considering syntax, we observed that 13 out of the 20 definitions generated demonstrated correct syntactic form and were overall comprehensible output sequences.

6 Conclusions

In this paper, we considered cross-lingual Wolastoqey-English definition modelling, in which we automatically generate English definitions for Wolastoqey words. Our work is in contrast to most prior work on definition modelling which has been monolingual, i.e., the word being defined and its definition are in the same language. In further contrast to most prior work on definition modelling, where Wolastoqey is a low-resource language, we do not have access to a large Wolastoqey background corpus for training. We showed that a sequence-to-sequence model that represents morphologically-complex Wolastoqey words at the sub-word level using BPE segmentation outperforms baseline approaches. We further demonstrated that the proposed approach can be improved by pretraining on small Wolastoqey and English monolingual corpora built from dictionary example sentences and by using a GRU instead of LSTM. Qualitative analysis revealed that the generated definitions are often syntactically well-formed and topically related to the gold-standard reference definitions.

In future work we plan to investigate alternative strategies for representing Wolastoqey words in the encoder, including character-level approaches and segmentations based on unsupervised approaches to learning morphology (Creutz and Lagus, 2002). We also plan to explore alternative model architectures, including transformer-based models (Vaswani et al., 2017) and models that incorporate large pre-trained English language models (e.g., Lewis et al., 2020).

Ethical Considerations

Wolastoqey is an Indigenous language and natural language processing can reinforce colonialist views

(Bird, 2020). The first author of this paper is Wolastoqew. The Passamaquoddy-Maliseet dictionary can be used for research purposes. We obtained permission to scrape the dictionary content for use in natural language processing research.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *Computing Research Repository*, arXiv:1409.0473.

Vidhisha Balachandran, Dheeraj Rajagopal, Rose Catherine Kanjirathinkal, and William Cohen. 2018. Learning to define terms in the software domain. In *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, pages 164–172, Brussels, Belgium. Association for Computational Linguistics.

Steven Bird. 2020. Decolonising speech and language technology. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3504–3519, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Steven Bird, Edward Loper, and Ewan Klein. 2009. *Natural Language Processing with Python*. O'Reilly Media Inc., Sebastopol, CA.

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.

Ting-Yun Chang and Yun-Nung Chen. 2019. What does this word mean? explaining contextualized embeddings with natural language definition. 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 6064–6070, Hong Kong, China. Association for Computational Linguistics.

Mathias Creutz and Krista Lagus. 2002. Unsupervised discovery of morphemes. In *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning*, pages 21–30. Association for Computational Linguistics.

Audrey Farber. 2015. A finite-state grammar of passamaquoddy-maliseet nouns. `http://dx.doi.org/10.13140/RG.2.1.2836.6967`.

David A. Francis and Robert M. Leavitt. 2008. *A Passamaquoddy-Maliseet Dictionary*. The University of Maine Press and Goose Lane Editions, Orono, United States and Fredericton Canada.

Artyom Gadetsky, Ilya Yakubovskiy, and Dmitry Vetrov. 2018. Conditional generators of words definitions. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 266–271, Melbourne, Australia. Association for Computational Linguistics.

Robert M. Leavitt. 1996. *Passamaquoddy-Maliseet*. Languages of the world. Materials, 27. LINCOM EUROPA, Munchen; Newcastle.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Timothee Mickus, Denis Paperno, and Matthieu Constant. 2019. Mark my word: A sequence-to-sequence approach to definition modeling. In *Proceedings of the First NLPL Workshop on Deep Learning for Natural Language Processing*, pages 1–11, Turku, Finland. Linköping University Electronic Press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Roberto Navigli and Paola Velardi. 2010. Learning word-class lattices for definition and hypernym extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1318–1327, Uppsala, Sweden. Association for Computational Linguistics.

Ke Ni and William Yang Wang. 2017. Learning to explain non-standard English words and phrases. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 413–417, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Thanapon Noraset, Chen Liang, Larry Birnbaum, and Doug Downey. 2017. Definition modeling: Learning to define word embeddings in natural language. In *AAAI 2017*, pages 3259–3266.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

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

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.

Statistics Canada. 2017. *Canada [Country] and Canada [Country] (table). Census Profile.* 2016 Census. Statistics Canada Catalogue no. 98-316-X2016001. Ottawa. Released November 29, 2017. https://www12.statcan.gc.ca/census-recensement/2016/dp-pd/prof/index.cfm?Lang=E (accessed August 13, 2021).

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, page 3104–3112, Montreal, Canada.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, Long Beach, California, USA.

Liner Yang, Cunliang Kong, Yun Chen, Yang Liu, Qinan Fan, and Erhong Yang. 2020. Incorporating sememes into chinese definition modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1669–1677.