Language Models for Code-switch Detection of te reo Māori and English in a Low-resource Setting

Jesin James[1], Vithya Yogarajan[2], Isabella Shields[1], Catherine Watson[1], Peter J Keegan[3], Peter-Lucas Jones[4] and Keoni Mahelona[4]

[1]Dept. of Electrical, Computer, and Software Engineering, The University of Auckland
[2]Strong AI Lab, School of Computer Science, The University of Auckland
[3]Te Puna Wānanga, The University of Auckland
[4]Te Hiku Media, New Zealand

Abstract

Te reo Māori, New Zealand's only indigenous language, is code-switched with English. Māori speakers are atleast bilingual, and the use of Māori is increasing in New Zealand English. Unfortunately, due to the minimal availability of resources, including digital data, Māori is under-represented in technological advances. Cloud-based multilingual systems such as Google and Microsoft Azure support Māori language detection. However, we provide experimental evidence to show that the accuracy of such systems is low when detecting Māori. Hence, with the support of Māori community, we collect Māori and bilingual data to use natural language processing (NLP) to improve Māori language detection. We train bilingual sub-word embeddings and provide evidence to show that our bilingual embeddings improve overall accuracy compared to the publicly-available monolingual embeddings. This improvement has been verified for various NLP tasks using three bilingual databases containing formal transcripts and informal social media data. We also show that BiLSTM with pre-trained Māori-English sub-word embeddings outperforms large-scale contextual language models such as BERT on down streaming tasks of detecting Māori language. However, this research uses large models 'as is' for transfer learning, where no further training was done on Māori-English data. The best accuracy of 87% was obtained using BiLSTM with bilingual embeddings to detect Māori-English code-switching points.

1 Introduction

Te reo Māori (referred to as Māori) is New Zealand's only indigenous language, spoken by 4.5% of the total population of 5 million. Māori speakers are bilingual, and code-switching between Māori and English is expected. Māori revitalisation efforts have increased Māori use in the otherwise English-speaking country. Detecting Māori language and code-switch instances is a prerequisite to analysing language data. Māori and English both use the Roman script (specifically, Māori uses a modified Roman script). Currently, annotations are done manually, making the process time-consuming and slowing down research and technology development. Consider the following sentences:

(a) Pērā anō i ngā mate kua hinga atu i te motu.
(b) I want to give no offence to my mate Willie Jackson, but once a week hardly qualifies as the significant Māori voice.

where green indicates Māori, red is used to indicate that the word has same spelling in Māori and English, and the remaining are English. Based on expert knowledge, we know the word mate is Māori in sentence (a) and English in sentence (b).

In this research, we focus on two primary tasks:

Task 1: Language Detection (LD) - detecting Māori language words from input text.
Task 2: Code-switch Detection (CS) - detecting Māori to English or English to Māori code-switch points from input text.

There is limited Māori-only and Māori-English bilingual data available. We collected data in collaboration with the Māori researchers, Māori technology developers and Māori community, where data-sharing is based on trust. As researchers, we remain guardians of the data, ensuring data sovereignty (Stats, 2020). Hence, all the resources shared from this study are bound by the Kaitiakitanga license (Te-Hiku-Media). This paper presents some of the first research to use advances in NLP to detect Māori language and code-switching. No existing models are using NLP techniques for Māori-English code-switch detection. Google and Microsoft Azure's cloud-based services are the only options available for language detection, which is the primary reason for using such large-scale multilingual cloud-based services for comparison in this paper.

This paper's contributions are:

1. Evaluation of detecting Māori using multilingual models, including the cloud-based services such as Google and Azure, and large scale language models such as Bidirectional Encoder Representations from Transformers (BERT).
2. Pre-training Māori-English bilingual, and Māori-only monolingual sub-word embeddings using collected data. Experiments using three different bilingual data for various NLP tasks show that bilingual embeddings outperform monolingual embeddings.
3. Large scale language models such as BERT –without further training on Māori-English data– fine-tuned on down streaming tasks of detecting Māori are outperformed by BiLSTM with fastText pre-trained sub-word bilingual embeddings for low-resourced language such as Māori.
4. Providing baseline results for detecting low-resourced Māori and code-switch between Māori-English language pair.

2 Te reo Māori (The Māori Language)

Māori is a Polynesian language belonging to the Austronesian family. Phonologically, Māori has ten consonants /p t k m n ŋ f r w h/. The Māori vowel system is described by five short vowels /i e a o u/ (Bauer et al., 1993). Orthographically, there is mostly a one-to-one mapping of a Māori phoneme to a grapheme, except for two digraphs, 'wh', which is /f/, and 'ng' which is /ŋ/. In modern orthography, long vowels are denoted with a macron (e.g. ā). In older texts, they are sometimes expressed as double vowels (e.g. aa), with an umlaut (e.g. ä), or ignored completely (that is, ā is written a). In addition, there is some regional variation in the way words are spelt (e.g. Aorangi vs Aoraki). English, in contrast, has a highly non-phonemic orthography. The Māori syllable structure consists of a nucleus, which may be occupied by a vowel (or a diphthong), and an optional onset (syllable start) occupied by a single consonant (consonant clusters are not present in Māori) (Harlow, 2007).

3 Related Work

Research using NLP for tasks relating to Māori is relatively young. Examples include statistical machine translation for Māori-English pair (Mohaghegh et al., 2014) and the inclusion of Māori language detection and translation using cloud services Google and Azure (Keegan, 2017). (Keegan, 2017) indicates that although the growth of cloud services for Māori translations is welcoming, due to the minimal availability of digitised Māori data, the resulting output is inaccurate. Google also acknowledges that for low-resource languages, the quality of language detection and automatic machine translation is far from perfect (Google-AI-Blog).

We present the first research that uses deep learning techniques to detect a code-switch between Māori and English. Hence, except for the Google and Azure cloud services (more details in Section 5.1), we are limited by the availability of systems for Māori language detection and Māori-English code-switch detection for comparison. We use approaches that were inspired by the literature on other language pairs. Examples include XNLI (Cross-lingual Natural Language Inference) cross-lingual classification benchmark (Conneau et al., 2018) where the bidirectional long short-term memory (BiLSTM) model was used across several low resource languages, including Swahili and Urdu; and code-switch detection using BiLSTM and Character-LSTM for language pair English-Hindi (Lal et al., 2019; Mukherjee et al., 2019). XNLI benchmark uses fastText common-crawl embeddings (denoted as E300 in this paper) and aligns it with the MUSE library. Comparison among deep learning models shows that adding background information through sub-word pre-trained embeddings trained using fastText and in the form of lexicons improves the overall performance of deep neural networks on databases of low-resource languages (Adouane et al., 2018).

Transformers such as BERT is the state-of-the-art in many NLP tasks, including language detection, named entity recognition, and machine translation (Devlin et al., 2019; Conneau et al., 2020). There are many large scale multilingual models, such as XLM-R (Conneau et al., 2020) and multilingual BERT (mBERT) (Devlin et al., 2019) trained in more than 100 languages. Research shows that for languages that are under-sampled during training, the effectiveness of large scale multilingual models such as mBERT are sub-optimal (Wu and Dredze, 2020; Wang et al., 2020). In comparison to the contextual representations such as BERT, embeddings with sub-word representation are more data-efficient when data availability is limited (Wu

Data	# Sentences	# Words	Text	Labels	Task
Hansard data[0]	2,021,261	36,757,230	formal	word-level & sentence level language labels	LD, CS
MLT corpus (Trye et al., 2019)	2,500	50,000	informal	tweet level labels: relevance/irrelevance	LD
RMT corpus (Trye et al., 2022)	79,018	1,000,000	informal	Māori words are identified and labelled	LD

Table 1: Databases used for experimental evaluations. LD: Language Detection, CS: Code-Switch Detection.

and Dredze, 2020). Furthermore, (Muller et al., 2021) provide evidence to show that many under-sampled or unseen languages during training –such as Maltese or Narabizi– code-mixed with French perform worse when using mBERT compared to an RNN with non-contextual dependency parsing baseline. It has been shown that for such unseen or under-sampled languages, there is a need to further train or fine-tune directly with available raw data in the unseen target languages (Muller et al., 2021).

4 Databases

Due to the low-resource nature of the Māori language, extensive databases are currently are unavailable. We collected text data from different sources to form the Māori-English Words (MEW) database, as summarised in Table 2. MEW database contains legal context, stories, social media posts and newspaper articles. The unlabelled MEW database is used to pre-train bilingual and Māori-only monolingual embeddings. We use three labelled databases for experiments: Hansard database, MLT corpus, and RMT corpus. Details of these databases are provided in Table 1.

Hansard database contains the New Zealand Parliament debates from 2003 onwards. Together with experts in Māori (Te-Hiku-Media), we have labelled the Hansard database, where English or Māori labels are assigned using linguistic rules and manual checking. Each sentence in the databases is marked as Māori, English or bilingual. Each word of each sentence is labelled as Māori or English. The resulting data includes 102,559 bilingual, 1,909,876 English-only and 8,826 Māori-only sentences.

The Māori Loanword Twitter (MLT) corpus is a small database, where each tweet is labelled as 'relevant' and 'irrelevant', based on the presence of a pre-determined set of Māori loanwords in a given tweet. Given detecting Māori language in tweets is a prerequisite to this task, we consider this task also as a Māori language detection task.

Name and Database	# Words
Māori only	
D1: Te Taka Database*[1]	9,862,131
D2: Nga Mahi corpus (James et al., 2020)	81,036
D3: Māori Wikipedia	431,280
D4a: LMC Corpus[2]	5,486,328
Total size of Māori-only database = 92 MB	
Māori and English	
D4b: LMC Corpus	7,197,059
D5: Niupepa (Māori Newspapers)[3]	5,050,988
D6: Twitter Corpus*(Trye et al., 2019)	48,289,375
Total size of bilingual data = 0.4 GB	

Table 2: Māori-English Words (MEW) database.'*' indicates private collections of data.

Reo Māori Twitter (RMT) corpus contains tweets, where at least 80% of text is in Māori. RMT corpus provides a list of 879,000 Māori words across the tweets. We use this corpus also for the language detection task where the aim is to detect the Māori words identified in the RMT corpus.

5 Language Models and Classifiers

This section provides details of the language models and classifiers we used. We evaluate the performance of cloud-based language detection systems from Google and Azure for Māori. We represent text as bag-of-words and sub-word embeddings using fastText. We use logistic regression and multinomial naive Bayes as baseline classifiers for language detection. We also use neural networks such as RNNs and CNNs to train and evaluate language detection and code-switch detection tasks. Furthermore, we fine-tune transformer models, BERT and mBERT, for the down streaming task of language detection.

[0] https://www.parliament.nz/en/pb/hansard-debates/rhr/

[1] Private collection of te reo Māori text data, Te Taka Keegan, The University of Waikato, New Zealand, 2021

[2] http://nzetc.victoria.ac.nz/tm/scholarly/tei-legalMaoriCorpus.html.

[3] http://www.nzdl.org/cgi-bin/library.cgi?a=p&p=about&c=niupepa.

5.1 Cloud-based Online Tools

Google Translate[4] and Microsoft Azure Cognitive Services language detection[5] are two popular cloud-based online tools that can detect multiple languages. Google supports 108 languages, including New Zealand English and Māori. Google's RNN-based GNMT (Google Neural Machine Translation) model (Wu et al., 2016) showed significant improvements in enabling translations to cover many languages, including low-resourced languages. Google recently replaced the GNMT model with a hybrid model (transformer encoder and RNN decoder). This model has shown significant improvements compared to the other machine translation systems. Azure's cognitive services can translate 100+ languages, including Māori. Azure's early-stage neural network model (Xiong et al., 2017) included a CNN-BiLSTM architecture. Recently, Azure has combined several machine learning algorithms and neural networks to provide various cognitive services.

5.2 Bag of words

Bag of words (BOW) is an effective method (Goldberg, 2017; Joulin et al., 2017) to represent text as a sparse vector, where the order of words in a document is not considered. The number of occurrences of a word or a binary value indicating that the word is present in the document is stored.

5.3 Word Embeddings

For language processing tasks, continuous word representations such as word embeddings trained on large unlabelled databases facilitate effective representation learning (Bojanowski et al., 2017; Joulin et al., 2016). Here, we use fastText (Bojanowski et al., 2017) to learn word embeddings, as novel words not present during training can also be represented using fastText-based embeddings. This can be beneficial for a low-resource setting. FastText supports two word embeddings models: continuous bag-of-words (CBOW) and Skip-grams (Mikolov et al., 2013). The CBOW predicts the specific word from the source context. Skip-gram predicts the source context from the specific word. The embeddings in this research are trained to the specifications of Wikipedia and common crawl fastText models (Grave et al., 2018) (referred to as E300) for both CBOW and Skip-gram[6]. E300 uses the CBOW method, character n-grams of length 5, window of size 5, and 10 negative samples per positive sample with 300 dimensions. The learning rate is 0.05. Table 3 provides details of our bilingual embeddings, which are available to on request, including E300 details for comparison.

Embeddings Model	Data	Size	# Unique Words
Monolingual Embeddings			
E300 (Grave et al., 2018)	downloaded	7GB	2,000,000
Māori-300/300SG	D1 - D4a	3GB	49,315
Bilingual Embeddings			
Model-Māori-Eng-300 (and 300SG)	D1 - D6	3GB	303,505

Table 3: Outline of fastText pre-trained 300 dimensional embeddings. The MEW database (Table 2) was used for training. 'SG': Skipgram model, otherwise it is CBOW.

5.4 Baseline Classifiers

We used multinomial naive Bayes (John and Langley, 1995) and logistic regression (LR) (Cox, 1958) to classify text features represented by BOW and static word embeddings. LR is a statistical model used to analyse databases where independent variables determine an outcome. Naive Bayes (John and Langley, 1995) is an easy to build supervised learning algorithm, which applies Bayes' theorem with the "naive" assumption of independence.

5.5 Convolutional Neural Network (CNN)

CNN for text (Kim, 2014) combines one-dimensional convolutions with a max-over-time pooling layer and a fully connected layer. If $x_{i:i+j}$ is a concatenation of words from a sentence, each word, x_i, x_{i+1}, ... is mapped to its k-dimensional embeddings using word embeddings. A new feature is produced using convolution. Max-over-time pooling is applied over the feature map to capture the most important feature value. The final prediction is made by computing a weighted combination of the pooled values and applying Softmax activation function.

5.6 Recurrent Neural Networks (RNN)

RNNs (Rumelhart et al., 1986) are designed to handle sequential data, such as text, where the

[4] https://translate.google.com/

[5] https://azure.microsoft.com/en-us/services/cognitive-services/translator/

[6] Embeddings trained on a 4 core Intel i7-6700K CPU @ 4.00GHz with 64GB of RAM. Average time: <30 minutes.

data contains complex temporal dependencies and hidden information. Long Short Term Memory networks (LSTM) (Hochreiter and Schmidhuber, 1997) are modified RNNs designed to overcome the issue of vanishing gradient with RNNs. LSTM has a gating mechanism consisting of input gate, forget gate, and output gate, ensuring a constant error flow and avoiding long-term dependency problems. The memory in LSTM is stored in an internal state, and the three gates play a vital role in deciding which information is to be included, added or removed from the memory. Over time, the memory cells learn which information is essential based on the weights. Bidirectional RNNs are widely used extensions where the input sequence is fed from beginning to end and from end to beginning. For BiLSTM (Grave et al., 2018), given there are two LSTM layers, the hidden layer output is split into two - for forward and backwards passes over the input.

5.7 Transformers

BERT (Devlin et al., 2019) is one of the early transformer models that apply bidirectional training of encoders (Vaswani et al., 2017) to language modelling. The 12-layer BERT-base model with a hidden size of 768, 12 self-attention heads, 110M parameter neural network architecture was pre-trained from scratch on BookCorpus and English Wikipedia. The mBERT-base (Devlin et al., 2019) model uses the same pre-training objective as BERT-base and is pre-trained with Wikipedia text of 104 languages with most articles. In this research, we use BERT and mBERT to refer to BERT-base and mBERT-base. It is vital to point out that this research does not pre-train BERT models (both BERT-base and mBERT-base) from scratch or continuously on the very limited available Māori language data. Instead, this research only performs fine tuning on down streaming task. There are evidence on mBERT performance of zero-shot tasks (Keung et al., 2020), and hence the decision to limit this study to only fine-tuning. Pre-training BERT models is out of scope of this current research.

6 Experimental Setup

We experiment with various language models and classifiers for two main tasks: language detection (LD) and code-switch detection (CS). Our ultimate goal is to find a combination of language modelling and NLP techniques to improve the overall accuracy of LD and CS tasks. We use three databases to evaluate these tasks with details provided in Table 1. We use the Hansard database sentences as the primary data for training and testing. All three datasets were pre-processed by lower-casing and using regular expressions to remove punctuation using Python 3.9 library with Pandas data frame. All experimental results are obtained from a random seeds training-testing scheme; 70% of the shuffled data is used for training, with 10% for validation and 20% for testing, and averaged over three runs. The variation of these three independent runs is within a range of ± 0.015.

To represent text we use both fastText pre-trained embeddings (see Table 3) and sparse vectors obtained from BOW representations. An overview of code-switch detection using trained models such as BiLSTM and CNN is presented in Figure 1. This diagram is an example to demonstrate the system we used for end-to-end code-switch detection using neural networks. Step 1 includes training and evaluating a neural network. We use the training set of the Hansard database to train the model and use validation loss as the stopping condition to avoid over-fitting. In step 2, we load the trained model and detect languages at the word level on testing data. Once the language detection is done, the points in the sentence where the language labels switch from Māori to English or from English to Māori are marked as code-switch points.

Neural network models presented in this research are implemented using Keras/Tensorflow. Adam (Kingma and Ba, 2015), an adaptive learning rate optimisation algorithm, is used as the optimiser for neural networks. Softmax activation function is used in the output layer of the network. We use a combination of dropout (Srivastava et al., 2014), with a rate of 0.5, and early stopping (Zhang et al., 2017) to avoid over-fitting. We use a maximum length of 250 tokens (or words) for BiLSTM and CNN, and padding for sentences with less than the maximum length. The term tokens and words is used interchangeably in this paper. The embeddings layer is with a dimension of 300. The hidden units of BiLSTM are 128, and the hidden units of one-dimensional convolutions are 128. For both CNN and BiLSTM, categorical cross-entropy is used as the loss function.

We also fine tune pre-trained transformers, BERT and mBERT on the down streaming task

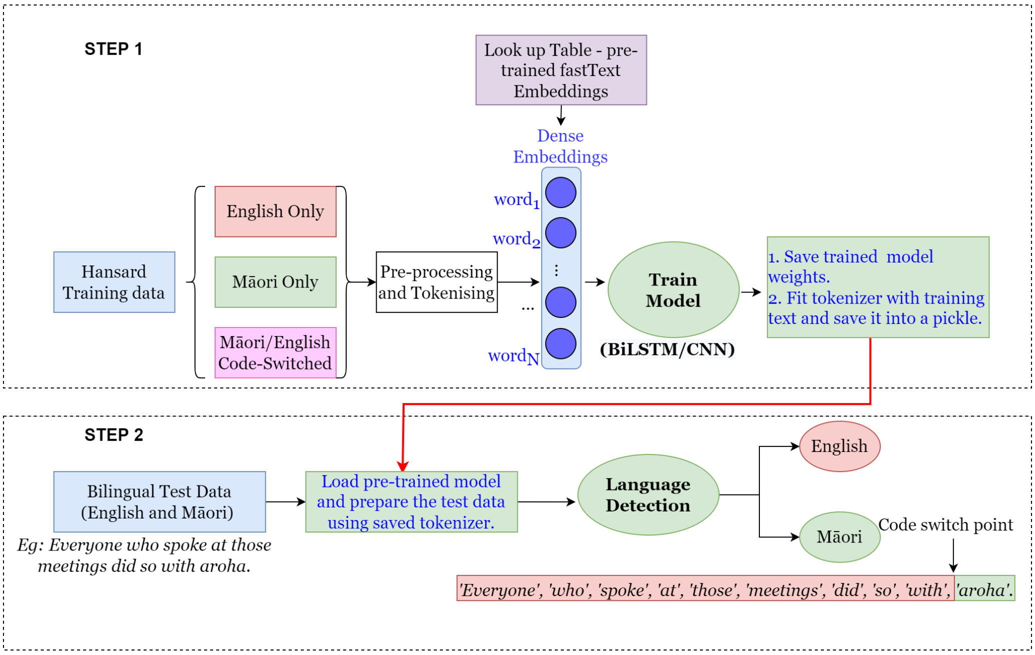
STEP 1
Look up Table - pre-trained fastText Embeddings
Dense Embeddings
word$_1$
word$_2$
word$_N$
Hansard Training data
English Only
Māori Only
Māori/English Code-Switched
Pre-processing and Tokenising
Train Model
(BiLSTM/CNN)
1. Save trained model weights.
2. Fit tokenizer with training text and save it into a pickle.
STEP 2
Bilingual Test Data (English and Māori)
Eg: Everyone who spoke at those meetings did so with aroha.
Load pre-trained model and prepare the test data using saved tokenizer.
Language Detection
English
Māori
Code switch point
'Everyone', 'who', 'spoke', 'at', 'those', 'meetings', 'did', 'so', 'with', 'aroha'.

Figure 1: Code-switch detection using neural networks. Example shows 'English' words {Everyone, who, spoke, at, those, meetings, did, so, with} are detected as 'English' and 'aroha' detected as 'Māori'.

of language detection. We use batch size of 16, maximum sequence length of 256 and learning rate of 1e-5. For both BERT and mBERT, the loss and accuracy were reported at each epoch. For both BERT and mBERT, the model converges fast, needing an average of 5 epochs per run.

All evaluations were done using Sklearn metrics[7]. Evaluations using baseline classifiers such as multilingual naive Bayes and LR with BOW and static features from embeddings require CPU only[8] machines and are very quick to train and evaluate. Neural networks require GPU devices[9] for efficient training and testing. The average training time for CNN was 150-180 minutes, and BiLSTM was 300-360 minutes, while BERT and mBERT required 240 minutes per epoch being trained for an average of 5 epochs. The testing time for trained deep learning models is rapid, requiring a few minutes. The code used in this research is made available[10].

We present overall macro-F1 score and weighted-F1 score to provide different insights (Toftrup et al., 2021; Khanuja et al., 2020). We also provide F1-scores of each label where appropriate. Macro-F1 provides average per-language results and is equally important to all languages. The weighted-F1 score considers the popularity of the languages in the data set.

The Nemenyi posthoc test (95% confidence level) identifies statistical differences between learning methods. Critical Difference (CD) plots show the average ranking of individual F1 scores obtained using various language models. The lower the rank, the better the model is. The difference in average ranking is statistically significant if there is no bold line connecting the two settings.

7 Experimental Results

The results are presented for the language detection (LD) tasks and code-switch detection (CS) tasks. The language detection task is a crucial first step for detecting code-switching (Rijhwani et al., 2017; Barman et al., 2014). First, we present the results of the language detection tasks using the three databases (Table 1), followed by the results of the code switch task using the Hansard database. As indicated in the experimental setup, all experimental results are obtained from a random seeds training-testing scheme and averaged over three runs. The variation of these three independent runs is within a range of ±0.015.

[7] https://scikit-learn.org/stable/modules/generated/

[8] 4 core Intel i7-6700K CPU @ 4.00GHz with 64GB of RAM.

[9] 12 core Intel(R) Xeon(R) W-2133 CPU @ 3.60GHz, GV100GL

[10] Pre-trained bilingual and monolingual embeddings are available for researchers on request. Experimental details, model implementations, and trained language models are available for researchers, all bound by the Kaitiakitanga license: https://github.com/MaoriEnglish-Codeswitch/MaoriEnglish-CodeSwitch-Detection

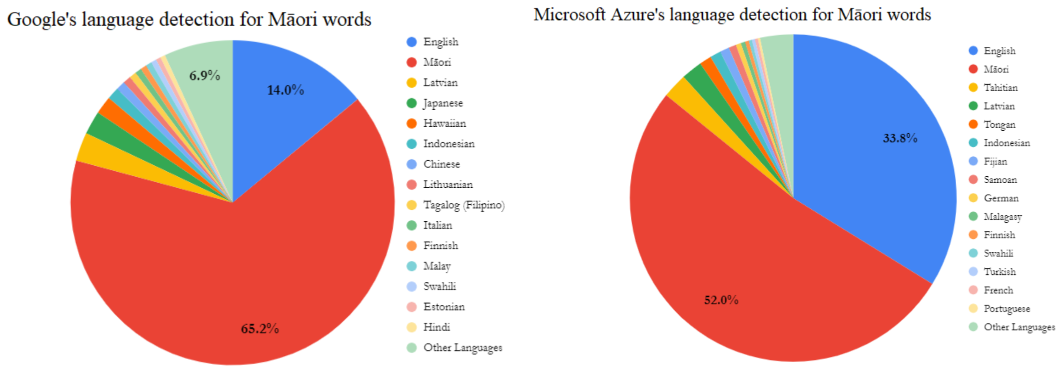

Figure 2: Pie Chart of the languages detected by Google (left) and Azure (right) at word level for the test set of the Hansard Database. The gold-standard label for all the words used here is ‘Māori’.

7.1 Task 1: Language Detection

7.1.1 Cloud-based Online Tools

To analyse the effectiveness of using Google Translate and Azure services to detect Māori (and English), we experimented with the test set of the Hansard database where the sentences are either monolingual (Māori or English) or code-switched. Google Translate detected 99.7% of the English words, and Azure detected 97.8% of the English words correctly. Figure 2 presents pie charts of the resulting language detection for ‘Māori’ word (i.e. the gold-standard labels for the words is ‘Māori’). For Māori words, Google Translate detected with an accuracy of 65.2%, and Azure detected with an accuracy of 52%. Although the accuracy of Google Translate was better than Azure, the error rate of both services are too high for Māori language detection. In addition, apart from wrongly detecting Māori words as English, around 14-21% of the words were classified as various other languages by both cloud services. We acknowledge that cloud-based services such as Google and Azure are multilingual and hence low-resource languages such as Māori are dominated by the resource-rich languages during training. This will inevitably influence the accuracy of LD of Māori using Google and Azure’s cloud services. However, given there is no other system available to detect Māori, it was still vital to evaluate the outcome of the above mentioned cloud-based services.

7.1.2 Baseline Classifiers

LD task using the Hansard database is a multi-class classification problem at the sentence level (classes: Māori, English or Code-Switched sentence). The LD task using MLT corpus is a binary classification problem of relevant/irrelevant tweets based on the usage of the Māori loanwords. Table 4 presents overall macro-F1 and F1 scores for the LD task using Hansard database and MLT corpus, respectively, where BOW and static word embeddings at the sentence level (or tweet level) are used to represent the text. We obtain embeddings for each sentence by computing the vector sum of the embeddings for each word in the sentence. This vector sum is then normalised to have length one, to ensure that sentences of different lengths have representations of similar magnitudes. The bilingual embeddings perform better than monolingual embeddings for both Hansard and MLT corpus. However, BOW outperforms static embeddings feature representation for LR.

Model	Data	Results
	Multi-class	Macro-F1
Multinomial NB (BOW)	Hansard	0.887
LR (BOW)	Hansard	**0.913**
LR (Eng300)	Hansard	0.831
LR (Māori-Eng-300)	Hansard	0.853
LR (Māori-Eng-300SG)	Hansard	0.859
	Binary	F1-score
LR (Eng300)	MLT corpus	0.833
LR (Māori-300SG)	MLT corpus	0.812
LR (Māori-Eng-300)	MLT corpus	**0.849**
LR (Māori-Eng-300SG)	MLT corpus	0.846

Table 4: Macro-F1 scores and F1-scores for the test set of Hansard database and labelled MLT corpus respectively, where BOW or sentence level features are used to represent text. **Bold**: best results for each task.

7.1.3 Neural Networks

After evaluating the performance of baseline classifiers, we further proceed with LD task using neural networks. As the size of the labelled MLT corpus is small, it is insufficient for training and evaluating neural networks. Table 5 presents macro-F1 and weighted-F1 scores obtained using the test set of the Hansard database for performance comparison

Model	Macro-F1	Weighted-F1
Monolingual Embeddings		
CNN (E300)	0.946	0.985
CNN (Māori-300)	0.905	0.986
CNN (Māori-300SG)	0.914	0.990
BiLSTM (E300)	0.943	0.996
BiLSTM (Māori-300)	0.926	0.995
BiLSTM (Māori-300SG)	0.940	0.995
Bilingual Embeddings		
CNN (Māori-Eng-300)	0.963	0.995
CNN (Māori-Eng-300SG)	0.969	0.996
BiLSTM (Māori-Eng-300)	0.984	**0.997**
BiLSTM (Māori-Eng-300SG)	**0.989**	**0.997**
Contextual Embeddings		
BERT-base	0.931	0.988
mBERT-base	0.946	0.991

Table 5: Comparison of results for the Hansard database (test set) with various models. **Bold**: best results.

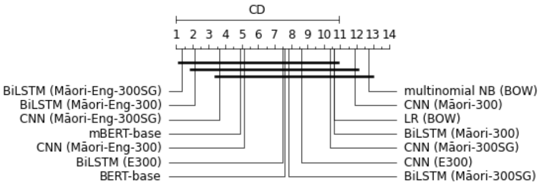

Figure 3: Critical difference plots identifying statistical differences between models presented in Tables 4 & 5.

Model	Training data	Testing data	Accuracy (Māori)
Google	Wikipedia	RMT	68.2%
BiLSTM (E300)	Hansard	RMT	56.6%
BiLSTM (Māori-Eng-300)	Hansard	RMT	85.4%
BiLSTM (Māori-Eng-300SG)	Hansard	RMT	**85.6%**

Table 6: Accuracy of Māori words detection in RMT corpus using Hansard-based trained models (Table 5).

across language models. The macro-F1 score is an unweighted average score of all the classes. In comparison, weighted-F1 scores are higher than macro-F1 scores across the models. The imbalanced distribution in the data, where labels are predominantly English, is reflected in the scores where the minority classes penalise the macro-F1 scores. Bilingual embeddings (Māori-Eng-300) consistently perform better than monolingual embeddings. BiLSTM with Māori-Eng-300SG embeddings are the best across all models, including BERT-base and mBERT-base. Skip-gram models are better than CBOW. In comparison, English-only embeddings E300 outperform Māori-only monolingual embeddings. One possible explanation for this is the lack of training data for Māori-only embeddings compared to E300.

Figure 3 presents critical difference plots across the models presented in Table 5 and BOW representation presented in Table 4. BiLSTM (Māori-Eng-300SG) has the lowest rank, and multinomial naive Bayes (BOW) has the highest rank with no bold line connecting the two, indicating the difference in average ranking is statistically significant. Bold lines are connecting BiLSTM (Māori-Eng-300SG) with mBERT and BERT-base in the CD-plot, indicating that the difference in average ranking is not statistically significant. A 4-6 % improvement was observed between BERT/mBERT and BiLSTM (Māori-Eng-300SG).

To further evaluate the language models, we used the models trained with the Hansard data to detect Māori words in the RMT corpus. Table 6 presents the accuracy of the detection. We also present the accuracy of Māori language detection using Google Translate for comparison. Evidently, BiLSTM with Māori-Eng-300SG embeddings model trained on the training set of the Hansard database has the best accuracy. As observed with other databases, the accuracy of the bilingual embeddings is higher than the monolingual embeddings. However, the accuracy of BiLSTM with E300 embeddings is considerably lower than other models, including Google. One possible reason is the lack of vocabulary in E300 for the informal language used in RMT data (Tweets).

7.1.4 In Summary

The results suggest that the bilingual embeddings perform better than monolingual embeddings (both the downloaded Eng300 and Māori only models) for the LD task. This finding was verified across the Hansard database (Tables 4, 5) and the MLT corpus (Table 4). Further evidence is provided in Māori words detection using RMT corpus (Table 6). We also observed that the bilingual embeddings outperformed the contextual embeddings. One possible reason for this finding is the lack of vocabulary in BERT models, as no further training was performed using Māori data. This research only fine tunes the BERT models for down streaming tasks. As emphasised before, the Māori data availability is the biggest limitation to this research. Among the experimented models for LD task, BiLSTM with Māori-Eng-300SG performed the best.

7.2 Task 2: Code-Switch Detection

For evaluation of the code-switch detection between Māori-English pair, we require word-level labels and hence, only the Hansard database was used for this task. We use selected trained models pre-

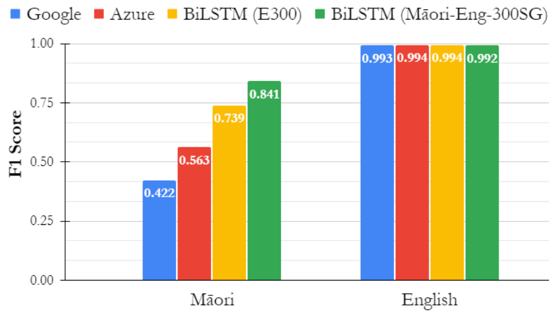

Figure 4: F1-scores for Māori and English calculated at the word level for the Hansard database.

Model	CS: Accuracy
CNN (E300)	35%
BiLSTM (E300)	83%
BiLSTM (Māori-Eng-300)	67%
BiLSTM (Māori-Eng-300SG)	**87%**

Table 7: Accuracy of code-switch detection in the Hansard data (bilingual sentences of the test set) using the trained models, as shown in Figure 1.

sented in Section 7.1, and identify the code-switch points (see Figure 1). Figure 4 presents word-level F1 scores of Māori and English for CS task. For English words, all systems perform equally well. However, for Māori, cloud-based multilingual systems perform poorly, and BiLSTM with bilingual embeddings shows a substantial improvement in F1 score, as observed before. It is vital to point out, cloud-based services such as Google and Azure are multilingual models and these systems have to classify between large number of languages. Hence, the poor performance with detecting Māori is not surprising, especially when compared to models which only have to classify between English and Māori. However, we include the results of large scale models here to emphasise the fact that the only existing tool that can detect Māori have limitations. Furthermore, Table 7 presents the accuracy of detecting the code-switch points of the test set of the Hansard database. Among the reported results, CNN with E300 performed poorly, and BiLSTM with Māori-Eng-300SG outperformed the other models.

8 Discussion and Conclusions

This research is the first attempt to use advances in NLP in two tasks - Māori (a low-resourced language) language detection, and Māori-English code-switch detection. Our experiments show that the accuracy of existing cloud-based systems to detect Māori is very low. Hence, there is the need to have more specialised systems for detecting Māori.

We collected data in collaboration with Māori researchers for training and evaluations. Experiments obtained across tasks using three databases show that our bilingual embeddings outperformed downloaded, pre-trained English-only embeddings trained on large databases. Among the models tested, BiLSTM with bilingual embeddings trained using the Skip-gram model is the best for both tasks. We provide evidence to show BERT-base used on the down-streaming task of language detection –where Māori is under-represented or unseen by the model vocabulary– is not always the best solution (as also observed by (Wu and Dredze, 2020; Wang et al., 2020)). For most low-resourced languages, including Māori, the Wikipedia data is significantly smaller than English, resulting in a reduced vocabulary. Due to limited resources, continuous training or training from scratch of models such as BERT-base is not possible.

For future work, it is a possibility to use ideas such as Extend M-BERT (Wang et al., 2020) and explore more efficient pre-training techniques to improve the accuracy of BERT like models for language detection of low-resource languages such as Māori. In addition, hybrid models using hand-crafted rules based on the phonotactic differences between the languages and deep learning-based methods are a promising pathway for future work.

The availability of digitised Māori and bilingual data is limited, which restricts the ability to train large language models. In addition, considering this is the first deep learning-based research in this area, comparison with published work is not possible. We overcome these limitations by respecting the available data and data sovereignty for this research. We provide experimental results using cloud services such as Google and Azure, as these are the only available systems that can detect Māori. The study reported here is a much-needed contribution to Māori language technology development. Word embeddings developed in this research are available to other researchers on request, bound by the Kaitiakitanga license.

Acknowledgements

Thanks to all researchers who shared their data collections with us, and to the University of Auckland Faculty Research Development Fund (FRDF) 2020 for support. We also thank Human Aspects of Software Engineering Lab (HASEL) at the University of Auckland for the computing facilities.

References

Wafia Adouane, Jean-Philippe Bernardy, and Simon Dobnik. 2018. Improving neural network performance by injecting background knowledge: Detecting code-switching and borrowing in algerian texts. In *Proc. of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 20–28.

Utsab Barman, Amitava Das, Joachim Wagner, and Jennifer Foster. 2014. Code mixing: A challenge for language identification in the language of social media. In *Proc. of the first workshop on computational approaches to code switching*, pages 13–23.

Winifred Bauer, William Parker, and Te Kareongawai Evans. 1993. In *Māori*. London: Routledge.

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.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proc. ACL Conference*.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating Cross-lingual Sentence Representations. In *Proc. EMNLP*, pages 2475–2485.

David R Cox. 1958. The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):215–232.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Yoav Goldberg. 2017. Neural network methods for natural language processing. *Synthesis Lectures on Human Language Technologies*, 10(1):1–309.

Google-AI-Blog. Google AI Blog: Recent advances in Google Translate. `https://ai.googleblog.com/2020/06/recent-advances-in-google-translate.html`, accessed Dec 15 2021.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proc. of the International Conference on Language Resources and Evaluation*, pages 3483–3487.

Ray Harlow. 2007. *Māori: A linguistic introduction*. Cambridge University Press.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Jesin James, Isabella Shields, Rebekah Berriman, Peter J. Keegan, and Catherine I. Watson. 2020. Developing Resources for Te Reo Māori Text To Speech Synthesis System. In *Proc. Sojka P., Kopeček I., Pala K., Horák A. (eds) Text, Speech, and Dialogue, Lecture Notes in Computer Science*, pages 294–302.

George H John and Pat Langley. 1995. Estimating continuous distributions in bayesian classifiers. In *Proc. Conference on Uncertainty in Artificial Intelligence*, pages 338–345.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proc. of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431.

Te Taka Adrian Gregory Keegan. 2017. Machine translation for te reo Māori. *He Whare Hangarau Māori - Language, culture & technology, Editors: Whaanga, H., Keegan, T. T., Apperley, M.. 23-28. Te Pua Wānanga ki te Ao / Faculty of Māori and Indigenous Studies, Te Whare Wānanga o Waikato / University of Waikato*, pages 23–28.

Phillip Keung, Yichao Lu, Julian Salazar, and Vikas Bhardwaj. 2020. Don't use english dev: On the zero-shot cross-lingual evaluation of contextual embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 549–554.

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

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proc. International Conference on Learning Representations (ICLR)*, pages 1–13.

Yash Kumar Lal, Vaibhav Kumar, Mrinal Dhar, Manish Shrivastava, and Philipp Koehn. 2019. De-mixing sentiment from code-mixed text. In *Proc. of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 371–377.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mahsa Mohaghegh, Michael McCauley, and Mehdi Mohammadi. 2014. Māori-English machine translation. *in Proc. NZCSRSC New Zealand Computer Science Research Student Conference, Canterbury University. Unitec Research Bank.*

Siddhartha Mukherjee, Vinuthkumar Prasan, Anish Nediyanchath, Manan Shah, and Nikhil Kumar. 2019. Robust deep learning based sentiment classification of code-mixed text. In *Proc. of the 16th International Conference on Natural Language Processing*, pages 124–129.

Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. In *Proc. of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462.

Shruti Rijhwani, Royal Sequiera, Monojit Choudhury, Kalika Bali, and Chandra Shekhar Maddila. 2017. Estimating code-switching on twitter with a novel generalized word-level language detection technique. In *Proc. of the 55th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 1971–1982.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1986. Learning representations by backpropagating errors. *Nature*, 323(6088):533–536.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

N. Z. Stats. 2020. Ngā tikanga paihere: a framework guiding ethical and culturally appropriate data use. *Guidelines*, page 8. https://data.govt.nz/toolkit/data-ethics/nga-tikanga-paihere/.

Te-Hiku-Media. Kaitiakitanga license. https://github.com/TeHikuMedia/Kaitiakitanga-License, accessed 10 Dec 2021.

Te-Hiku-Media. Te Hiku Media. https://tehiku.nz/te-hiku-tech/papa-reo/.

Mads Toftrup, Søren Asger Sørensen, Manuel R. Ciosici, and Ira Assent. 2021. A reproduction of apple's bi-directional LSTM models for language identification in short strings. In *Proc. of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 36–42.

David Trye, Andreea S Calude, Felipe Bravo-Marquez, and Te Taka Adrian Gregory Keegan. 2019. Māori loanwords: a corpus of New Zealand English tweets. In *Proc. of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 136–142.

David Trye, Te Taka Keegan, Paora Mato, and Mark Apperley. 2022. Harnessing indigenous tweets: The Reo Māori Twitter corpus. *Language resources and evaluation*, pages 1–40.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30:5998–6008.

Zihan Wang, K Karthikeyan, Stephen Mayhew, and Dan Roth. 2020. Extending multilingual bert to low-resource languages. In *Proc. of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 2649–2656.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proc. of the 5th Workshop on Representation Learning for NLP*, pages 120–130.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig. 2017. The microsoft 2016 conversational speech recognition system. In *Proc. Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 5255–5259.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2017. Understanding deep learning requires re-thinking generalization. In *Proc. International Conference on Learning Representations 2017*, pages 1–15.