# Transformer-based Text Generation for Code-Switched Sepedi-English News

Simon Phetole Ramalepe[1[0000−0001−5660−8099]], Thipe I. Modipa[1,3,4[0000−0002−5383−5808]], and Marelie H. Davel[2,3,4[0000−0003−3103−5858]]

[1] Department of Computer Science, University of Limpopo, South Africa
simon.ramalepe@ul.ac.za, thipe.modipa@ul.ac.za
[2] Faculty of Engineering, North-West University, South Africa
[3] Centre for Artificial Intelligence Research, South Africa
[4] National Institute for Theoretical and Computational Sciences, South Africa

**Abstract.** Code-switched data is rarely available in written form and this makes the development of large datasets required to train code-switched language models difficult. Currently, available Sepedi-English code-switched corpora are not large enough to train a Transformer-based model for this language pair. In prior work, larger synthetic datasets have been constructed using a combination of a monolingual and a parallel corpus to approximate authentic code-switched text. In this study, we develop and analyse a new Sepedi-English news dataset (SepEnews). We collect and curate data from local radio news bulletins and use this to augment two existing sources collected from Sepedi newspapers and news headlines, respectively. We then develop and train a Transformer-based model for generating historic code-switched news, and demonstrate and analyse the system's performance.

**Keywords:** Code-switched data · Transformer model · Text generation.

## 1 Introduction

Code-switching is the practice of alternating between two languages in a discourse [23]. This phenomenon is generally practised in multilingual countries with more than one official language where speakers alternate words, phrases or sentences in a conversation [7]. It is a speaker- and domain-dependent, driven by factors like context, environment, and personal preferences [6]. Although code-switching is more prevalent in automatic speech recognition (ASR) systems, it has recently become more common in natural language processing (NLP) systems. Conversational systems like chatbots, sentiment analysis and question-and-answering systems allow multilingual individuals to communicate seamlessly as they would in a normal setting. Code-switching still occurs mostly in a spoken rather than in a written context, hence, it is challenging to collect enough text-based code-switched data for training language models. Various modelling approaches have been investigated to generate code-switched data, including the use of recurrent neural networks (RNNs) and long-short-term-memory (LSTM) networks [11], generative adversarial networks (GANs) [2]; [6] and variational autoencoders for code-switched text (VACS) [29]. These approaches can generate plausible monolingual text but struggle to model long-term dependencies. In other words, they tend to forget the context as the sentence gets longer. The advent of the Transformer-based architecture introduced by Vaswami *et al.* [30] has made it possible to develop language models that can remember and maintain context through the use of multi-head self-attention mechanisms. However, the performance of these models on code-switched data still tends to be poor due to the paucity of code-switched data for training [29].

The purpose of this study, therefore, is to develop and analyse a new domain-specific non-synthetic code-switched corpus for the Sepedi-English language pair from the news domain and subsequently develop a Transformer-based text generation model. We use radio news bulletins (in their written format) as the main source material. This study focuses on the text generation model itself, with the ultimate aim of developing a speech-enabled Sepedi system that can generate historic news and answer news-related questions in a code-switched manner. The paper is organized as follows: Section 2 discusses the background of the study. In section 3, we discuss the approaches we used for data collection and model development. We then analyse the results in section 4 before we give a conclusion in section 5.

## 2 Background

Collecting large volumes of code-switched text data from speech data is difficult and laborious due to the extensive transcription process that is required [24]. Examples of such collection efforts include, Hamed *et al.* [8], who collected spontaneous Egyptian Arabic-English code-switched speech data by conducting and recording informal interview conversations. The data was transcribed to generate a code-switched text dataset to train code-switched language models. In Hamed *et al.* [7], the authors collected code-switched data from online documents using a locally developed web spider program and a Bing web search engine. The collected data had about 2.3M sentences. 12.6% of the data collected from online libraries contained code-switched sentences and only 2.5% of the data collected from Bing search contained code-switched sentences.

In an alternative approach, multilingual code-switched corpora for English-isiZulu, English-isiXhosa, English-Setswana, and English-Sesotho were developed from recordings of soap operas [31]. Data for this multilingual code-switched corpus was obtained from digital video recordings of 626 South African soap opera episodes. The speech data was manually transcribed by fluent bilingual speakers of each language pair, and the transcriptions were used to generate the code-switched datasets. The most code-switching trigger words were identified in each of the language pairs.

Modipa and Davel [19], developed a Sepedi-English code-switched speech corpus to analyse the effect of Sepedi-English code-switching on automatic speech recognition (ASR) systems. In their data collection approach, radio broadcasts were recorded to create the Sepedi Radio (SR) dataset. As part of a pronunciation modelling study, the SR dataset was then transcribed and analysed and the phrases that contained code-switching instances were used as prompts to create a new dataset, referred to as the Sepedi-prompted code-switched corpus (SPCS). The process used to create these datasets is discussed in [18]. The SPCS dataset contains short code-switched phrases which make it impractical for text generation of longer sentences. Another effort to collect and analyse the Sepedi-English code-switched data was performed by Ramalepe *et al.* [27], however, the dataset was too small to effectively train a Transformer-based model for text generation.

### 2.1 Text generation approaches

Text generation is described as the process of automatically generating readable text given some input text; this is typically achieved by training a language model on a large corpus [14]. Here we describe different approaches to text generation for both code-switched data and monolingual data.

In their attempt to build a code-switched language model for the Arabic-English language pair, Hamed *et al.* [7] used the SRI Language Modeling Toolkit to develop a code-switched language model. First, they developed a baseline language model by interpolating model weights of the existing language models for each language. As discussed in Section 2, they collected data from online libraries and search engines. Their collected data was divided into three datasets (monolingual Arabic text, monolingual English text, and code-switch text). They used these datasets to train three language models. Model weights of each language model were interpolated and the best code-switched language model recorded a perplexity of 275.41 compared to 11841.9 perplexity from the baseline model.

In a study by Buzea *et al.* [1], the authors developed a text generation model for the Romanian language using a standard GPT-2 architecture. The model was trained on a small dataset of 24k news items crawled from online news portals. The authors evaluated the quality of the generated text using automatic evaluation metrics and their model was equally comparable to a larger Romanian model. They used a prompt of 100 words as input to the model during training and set it to generate text with a minimum of 600 words to a maximum of 4000 words.

Key and Cheng [12] used the Bidirectional Encoder Representations from Transformers (BERT) to initially develop a model for personality classification and personality-specific language generation using monolingual English data. They scraped their data from PersonalityCafe's Myers-Briggs Type Indicator (MBTI) forums and considered posts with 50 characters or more.

In another study, Du *et al.* [3] used both an RNN and the Generative Pre-Trained (GPT-2) approach to develop a model for providing large-scale support for online learning communities. Both the RNN and GPT-2 were trained using 2M comments collected from the Scratch[5] online community.

[5] https://scratch.mit.edu/

We are only aware of two monolingual Sepedi text generation systems that have been developed to date. The National Centre for Human Language Technology (NCHLT) corpus [25] for Sepedi was used to train an LSTM model for text generation [20]. The corpus was created from a collection of several South African government entities crawled from gov.za websites from 2007 to 2011. The Sepedi text corpus has 69K rows of text with about 2.1M tokens. Ramalepe *et al.* [28] used the same corpus to develop a Sepedi Transformer-based model.

Table 1 lists prominent text generation approaches, with English and Sepedi examples. The techniques listed in Table 1 report on performance using cross-entropy loss and top-k accuracy: the number of correct words in the top k predictions, as a percentage of the total number of words in the reference text. (When $k$ is 1, this metric becomes standard accuracy.) We provide these values as a broad indication of performance. However note that parameters such as the length of the seed text, or the maximum length of the generated text have a large effect on performance metrics, and differ across systems.

**Table 1.** Prominent text generation approaches and reported performance on the task initially developed for.

| **Technique** | **Language** | **Evaluation** | **Score** |
|---|---|---|---|
| BERT [12] | English | Accurracy | 47.0% |
| | | Loss | 0.02 |
| Transformer [27] | Sepedi-English | Accuracy | 66.0% |
| LSTM [20] | Sepedi | Accurracy | 50.3% |
| GPT [28] | Sepedi | Accurracy | 75.5% |
| | | Loss | 1.02 |

## 2.2 Transformers

Transformer-based models have emerged to outperform older recurrent architectures in modelling many natural language tasks, including text generation. These models use a multi-head self-attention mechanism to attend to different positions in the input sequence. The mechanism computes contextual representations of each word by considering the entire input sequence at different subspaces [33]. This technique enables the model to capture long-term dependencies which emerged as a challenge in conventional models.

Generative pre-trained language models like GPT-2 [26] (small, medium and large) use the decoder part of this Transformer architecture by stacking multiple Transformer decoder layers with masked self-attention heads. These language models are trained to predict the next word $x_i$ given the previous words $x_1, x_2..., x_{i-1}$ in the corpus (X) with the training objective of maximising the log-likelihood:

$$L_1(X) = \sum_i log(P(x_i|x_1, x_2, ..., x_{i-1})); \theta T) \quad (1)$$

where $\theta$T are the model parameters [17]. Like the GPT models, we also leverage the decoder part of the Transformer architecture in this study.

# 3 Approach

We approach the Sepedi-English text generation task by first creating a new dataset, as described in the next section. The new dataset is used to train a Transformer-based model, using the experimental setup of Section 3.2. In order to perform a code-switched analysis, individual words are classified as English or Sepedi, as discussed in Section 3.3.

## 3.1 Data collection

Since the available Sepedi-English corpus is not large enough to train a Transformer-based model, three sources of data were used to develop a new dataset. All these sources were found to contain instances of code-switching:

- Local news dataset: data for this dataset was collected from a local community radio station. Instead of recording and transcribing, we used the news bulletins created before recording from the station. Several broadcast shows are presented daily. Our data collection focused on radio news read during the various times of the day between June and October 2022. Permission to collect data was granted by the radio station.
- Sepedi newspaper dataset: the data for this dataset was collected from Sepedi newspapers between 2011 and 2022 and it is freely available for research purposes [15]. We analysed this dataset for the presence of code-switching instances and used it to augment the local news dataset.
- News headlines dataset: the data for this dataset was collected from the news headlines of a national radio station. It was created by Marivate *et al.* [16] for news classification. News headlines that are published as posts were scrapped from the radio station's social media page. The dataset is freely available for

**Table 2.** Examples of code-switching in the SepEnews dataset. The English version in italics has been manually translated

| |
|---|
| Gosasa mohlatša tona ka kgorong ya tša maphelo Dr Sibongiseni Dlomo o ya go bula semmušo **forum** ya ngwaga ka ngwaga ya bone ya go tla ka maano ago lwantšhana le bolwetši bja Malaria ka Cape Town. **Forum** yeo ya matšatši a mararo e ya go sepetšwaka tlase ga moeno woo orego **Accelerate Elimination to achieve global technical strategy milestones** |
| *Tomorrow Deputy Minister of the Department of Health Dr Sibongiseni Dlomo is going to officially open the fourth annual forum to discuss plans to fight against Malaria in Cape Town. The three-day forum will be run under the theme that says AccelerateElimination to achieve Global technical strategy milestones.* |
| go ya leka SAPU **unit** yeo sale e tswalelwa<br>*according to SAPU that unit has since closed* |

research purposes. Analysis of this dataset for the presence of code-switching instances has been conducted in Ramalepe *et al.* [27].

The matrix language in all these datasets is Sepedi, with English as a secondary language. We combined the three datasets to create a new dataset referred to as the Sepedi-English news (SepEnews) dataset.

Data cleaning was performed on all the datasets to standardise the data. We remove the names of the news readers, the editor, date, time slots, repeated full stops, forward and backward slashes and unnecessary blank spaces. We then preprocess the data by splitting it into tokens. (We used the TextVectorization layer from Keras for both tokenization and padding.) We show an example of code-switching in the SepEnews dataset in Table 2.

### 3.2 Model development

We develop and train a transformer-based model for the generation of code-switched news for the Sepedi-English language pair, referred to as the SEC-T model from here onwards. We opt for an approach similar to GPT-2 [26], since it has shown great success in text generation [32]. Specifically, we follow the process as defined by Landup [13] for developing a Transformer model. We also use the decoder part of the Transformer architecture with two separate embedding layers for tokens and a token index, with one dense layer at the top of the decoder.

We use the token and position embedding layer to encode the input data and pass the vocabulary size, maximum length and embedding dimension to it. The embedding layer is then passed as input to the decoder of the Transformer. To determine the output (next word in the sentence) we use a dense layer on top of the Transformer block with the rectified linear unit (ReLU) activation function. The developed model has three Transformer blocks with causal masking on the attention layers and four attention heads. In order to optimise the model hyperparameters, we use a Bayesian hyperparameter optimisation algorithm, as implemented by Keras [6]. To obtain the optimal hyperparameters for the model, a new set of hyperparameter values is generated in each trial. The values are then used by the tuner to fit and evaluate the model until a good set of hyperparameters is obtained. The best hyperparameters were then obtained from the tuner and used to configure our model. A standard cross-entropy loss function is used and Adam is used as the model optimiser. Early stopping on the validation loss is used to both regularise the model and to ensure that training does not continue beyond convergence. The minimum number of epochs is initially set at 10 and the maximum at 50 (which could be extended if early stopping ever selected the final training iteration as the best result). The learning rate range is searched for between 1e-01 and 1e-05 and the dropout between 0.1 and 0.4.

To train the model, we split the dataset as follows: 70% training, 10% validation and 20% testing. The test set is used as a holdout set for testing the performance of the model.

We use categorical cross-entropy loss and accuracy to measure performance.

[6] https://keras.io/keras_tuner/

### 3.3 Word classification

In order to analyse English and Sepedi results separately, it is necessary to identify and create a word list for each language. For this purpose, we develop a word classification process that relies heavily on a dictionary lookup technique. A similar approach was used by Nguyen and Dogruo [21], Gambäck [5], and Nguyen *et al.* [22].

We start by identifying common nouns using the named entity recognition (NER) list developed by Eiselen [4]. We use the NCHLT dictionaries for Sepedi and English, to classify individual words in the SepEnews dataset as either Sepedi, English, common to both languages or other (words from other languages). Unidentified words from this process are classified as unknown words.

To determine the number of English words in the dataset, we further develop and train a custom language identification (LID) tool using Sepedi and English monolingual data from NCHLT to classify words from the unknown list. We regard unknown words identified as English at this point as 'probably English' (these words are not yet verified as such) and words from the dictionary lookup method as 'definitely English'.

## 4 Analysis and Results

We first analyse the developed dataset before describing the training process of the SEC-T model. We then evaluate the performance of the model both in general and from a code-switching perspective.

### 4.1 Dataset analysis

Identifying individual words of a language with only limited language resources (as in the case of Sepedi) is a complex process. At this point, we followed a fully automated process as a preliminary step towards word identification. Further verification of this process is planned but is currently considered as future work. The SepEnews corpus consists of approximately 144K tokens, as shown in Table 3. Using the dictionary lookup approach and the LID technique, we record the number of unique English words as 1 034 which is 11.34% of the total unique words in the dataset. However, the distribution of these words in the dataset is low at 3.15%, which indicates that there is a limited amount of code-switching in the dataset.

Of the 1 093 unique English words, 888 words are regarded as 'definitely English' words (the rest as 'probably English'. Although the number of common words in the dataset is low at 0.59%, manual analysis of these words suggests that identification and classification of common words based on context is needed. In Table 4 it can be seen that unique proper nouns contribute just 12.19%. Words from other languages contribute 1.54% (we accept these words without further analysis). Sepedi words account for 71.98% of the total unique words in the dataset.

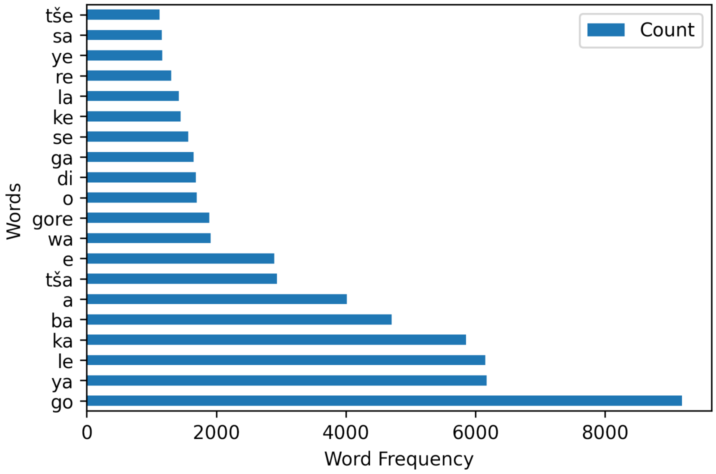

**Fig. 1.** The 20 most frequent words in the SepEnews dataset.

In Figure 1, we show the top 20 most common words in the SepEnews dataset. The x-axis shows the word frequency while the y-axis lists the words. In Figure 2, we show the frequency distribution of the 3 most common English words in the SepEnews dataset. The x-axis indicates the dataset partitioned into equal-sized segments.

### 4.2 Model performance

The optimal model's loss and accuracy curves are shown in Figure 3, as calculated at each epoch during the training process. Early stopping helped to control overfitting during training which was observable towards the last 10 epochs. The curves indicate that an asymptote has been reached before 50 epochs. Table 6 shows the optimal hyperparameters selected for the final model analysed. There were 2.7 million trainable model parameters.

Table 5 shows the results for the SEC-T model, with $k$ set to 40. The model obtained an optimal validation loss of 1.58 with a test accuracy of 77.6%. This

**Table 3.** The number of sentences, words, unique words, English words, and percentage of English words in the three datasets that together form SepEnews.

| Dataset | # Sentences | # Total words | # Unique words | # English unique words | % English unique words |
|---|---|---|---|---|---|
| Local News | 900 | 22 357 | 2 452 | 281 | 11.46% |
| Headlines News | 1 182 | 16 135 | 2 781 | 176 | 6.33% |
| Sepedi Newspaper | 3 108 | 106 279 | 7 320 | 849 | 11.60% |
| **SepEnews** | **5 190** | **144 358** | **9 638** | **1 093** | **11.34%** |

**Table 4.** Results of the individual words classification of the SepEnews dataset

| | Unique Sepedi | Unique English | Nouns | Common | Other | Unknown |
|---|---|---|---|---|---|---|
| # words | 6 938 | 1 093 | 1 175 | 57 | 149 | 334 |
| % of total words | 71.98% | 11.34% | 12.19% | 0.59% | 1.54% | 3.5% |

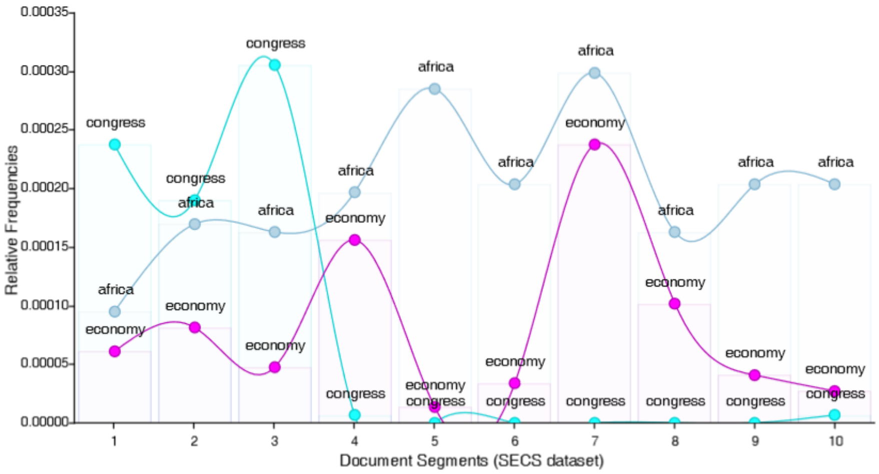

**Fig. 2.** Frequency distribution of the 3 most common English words in the SepEnews dataset

is a 9.0% absolute improvement over the model developed in [27], which is from the same domain (code-switched news). SEC-T's performance is also a 2.7% absolute improvement in accuracy compared to the Transformer-based model developed in [28] using a monolingual Sepedi dataset, an easier task.

In order to better understand the impact of the new data versus the impact of the model, the model from [27] was retrained, using the exact same architecture and protocol as described in [27]. This produced an improvement of 2.7% from 66.0% to 68.7%. It is clear that the updated dataset in itself is not as useful as it becomes when the architecture is adjusted to better utilise the additional data.

**Table 5.** Validation loss and test accuracy of the model from [27], retrained on the SepEnews dataset and the SEC-T model developed here.

| **Model** | **Dataset** | **Val loss** | **Test acc** |
|---|---|---|---|
| [27] retrained | SepEnews | 3.54 | 68.6% |
| SEC-T | SepEnews | 1.58 | 77.6% |

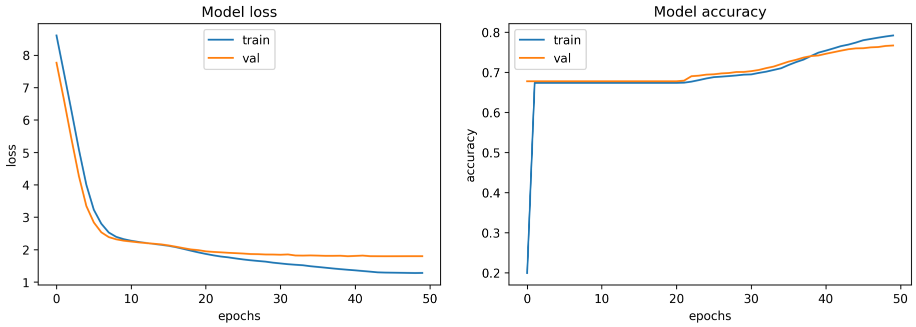

**Fig. 3.** Loss and accuracy curve

**Table 6.** Optimal model hyperparameters.

| Hyperparameter | Training value |
|---|---|
| Transformer layers | 3 |
| Embedding size | 100 |
| Attention heads | 4 |
| Dropout rate | 0.36 |
| Learning rate | 1e-03 |
| Batch size | 128 |

### 4.3 Code-switching analysis

We now analyse the frequency and accuracy of the generated code-switched words. To generate the text, seed text of length 6 is provided and an additional maximum of 80 words are generated. Although top-p or Nuclear sampling has emerged as the preferred decoding technique [9], top-k provides a competitive alternative where the next top-k possible tokens are sampled according to their relative probabilities at each time-step [10].

The top-k search samples the next token to be generated from a fixed number of tokens $k$, where $k$ is a non-negative integer value significantly less than the vocabulary size, and reported on when providing results. Compared to other decoding algorithms like beam search, top-k generates a considerably higher quality text [9]. We gradually increase the maximum length of tokens to be generated (from 40 to 80) while keeping the seed fixed to observe the number of tokens generated by the model. We recorded the number of Sepedi words versus English words that were generated at each iteration. For each maximum length, we computed the averages of the generated words together with the test accuracy from three iterations. Results (excluding the prompt) are shown in Table 7. Unlike in 3, we show the total number of words generated for each language.

**Table 7.** Number of English and Sepedi tokens generated, as the maximum length of tokens to be generated is varied.

| **Max length** | **#generated words** | **#Sepedi words** | **#English words** | **Other words** | **accuracy** |
|---|---|---|---|---|---|
| 40 | 32 | 26 | 5 | 1 | 77.5% |
| 60 | 43 | 26 | 16 | 1 | 71.6% |
| 80 | 59 | 41 | 18 | - | 63.7% |

We note that as the number of words in the generated text increases the number of English words (including potentially English words) also increases. Words that are neither Sepedi nor English are very few in the generated text. The last column shows the average accuracy for each maximum length. We also note that although the maximum length is never reached, the model does generate sentences that are grammatically correct.

### 4.4 Discussion

Although the number of unique English words in the datasets was small (1 093), the model could still generate English words that were grammatically in agreement with the matrix language, for example *le go thoma today* (and to start today) and *mo lebakeng le go support bana* (in this time and to support the kids). The frequency of code-switching from the generated text is still low, as expected given the training data. However, the ability of the model to utilise the current corpus to generate such code-switched text is promising.

**Table 8.** Generated text

| |
|---|
| **Thobela, ditaba di re mopresidente wa maloba wa lekala la covid-19, gomme ge re na le seemo sa kgone go tšea karolo ye bohlokwa kudu, le go thoma today a tla fa bohlatse le ge mpa ka mo lebakeng le go support bana ba le go tšwa le go netefatša gore le go ya maphelo ya ditirelo le le gore bontši le bana ba tla tša gore ba bangwe** |
| *Greeting, the news says that the former president of covid-19, and when we have a situation that we cannot take an important part and start today will give evidence if in this opportunity will support children and again to make sure that action health will many children and others.* |

Table 8 shows an example of generated text with a manually translated English version of the text. While the text is grammatically correct, it is not yet semantically useful.

For a Transformer-based model, a large training dataset is required and it remains our goal to acquire more code-switched data to improve the model performance.

## 5 Conclusions

The focus of this study was to develop and analyse a Sepedi-English code-switched corpus from the news domain and to develop a Transformer-based model for the generation of historical news. We discussed the process used to develop a new SepEnews dataset and analysed the dataset for the presence of code-switching. We note that although the presence of words common to both languages in the dataset is very low,

these words are challenging and can make it difficult to properly analyse individual words.

We used the developed dataset to train a Transformer-based model and achieved the highest test accuracy reported to date for this language pair (2.6% absolute higher than the accuracy obtained in [28] on a monolingual dataset, when using the same evaluation protocol). Future work includes continuing to grow the size of the dataset, exploring alternative approaches for data collection, and using human annotators to tag and evaluate different aspects of the system. This will allow us to establish a monolingual baseline against which approaches that utilise pretrained models from more highly resourced languages can be evaluated. We also aim to develop more sensitive, context-aware language identification systems, in order to be able to better differentiate between possible and true instances of code-switching.

## References

1. Buzea, M.C., Trăușan-Matu, , Rebedea, T.: Automatic Romanian text generation using GPT-2. UPB Scientific Bulletin **84**(4) (2022)
2. Chang, C.T., Chuang, S.P., Lee, H.Y.: Code-switching Sentence Generation by Generative Adversarial Networks and its Application to Data Augmentation (Jun 2019), arXiv:1811.02356 [cs]
3. Du, H., Xing, W., Pei, B.: Automatic text generation using deep learning: providing large-scale support for online learning communities. Interactive Learning Environments pp. 1–16 (Oct 2021)
4. Eiselen, R.: Government Domain Named Entity Recognition for South African Languages. In: In Proceedings of the Tenth International Conference on Language Resources and Evaluation. pp. 3344–3348 (May 2016)
5. Gambäck, B.: On measuring the complexity of code-mixing. In: InProceedings of the 11th international conference on natural language processing, Goa, India1. pp. 1–7 (Dec 2014)
6. Gao, Y., Feng, J., Liu, Y., Hou, L., Pan, X., Ma, Y.: Code-switching sentence generation by Bert and Generative Adversarial Networks. In: Interspeech 2019. pp. 3525–3529. ISCA (Sep 2019)
7. Hamed, I., Elmahdy, M., Abdennadher, S.: Building a first language model for code-switch Arabic-English. Procedia Computer Science **117**, 208–216 (Jan 2017)
8. Hamed, I., Elmahdy, M., Abdennadher, S.: Collection and Analysis of Code-switch Egyptian Arabic-English Speech Corpus. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). European Language Resources Association (ELRA), Miyazaki, Japan (May 2018)

9. Holtzman, A., Buys, J., Du, L., Forbes, M., Choi, Y.: The Curious Case of Neural Text Degeneration (Feb 2020)
10. Holtzman, A., Buys, J., Forbes, M., Bosselut, A., Golub, D., Choi, Y.: Learning to Write with Cooperative Discriminators. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1638–1649. Association for Computational Linguistics, Melbourne, Australia (Jul 2018)
11. Islam, M.S., Sharmin Mousumi, S.S., Abujar, S., Hossain, S.A.: Sequence-to-sequence Bangla Sentence Generation with LSTM Recurrent Neural Networks. Procedia Computer Science **152**, 51–58 (Jan 2019)
12. Keh, S.S., Cheng, I.T.: Myers-Briggs Personality Classification and Personality-Specific language generation using Pre-trained language models (Jul 2019)
13. Landup, D.: 5-Line GPT-Style Text Generation in Python with TensorFlow/Keras (Jun 2022), https://stackabuse.com/gpt-style-text-generation-in-python-with-tensorflowkeras/
14. Li, J., Tang, T., Zhao, W.X., Wen, J.R.: Pretrained language models for text generation: A survey (May 2021)
15. Marivate, V., Njini, D., Madodonga, A., Lastrucci, R., Dzingirai, I., Rajab, J.: The Vuk'uzenzele South African multilingual corpus (Feb 2023)
16. Marivate, V., Sefara, T., Chabalala, V., Makhaya, K., Mokgonyane, T., Mokoena, R., Modupe, A.: Investigating an approach for low resource language dataset creation, curation and classification: Setswana and Sepedi (Feb 2020)
17. Min, B., Ross, H.H., Sulem, E., Veyseh, A.P.B., Nguyen, T.H., Sainz, O., Agirre, E., Heinz, I., Roth, D.: Recent Advances in Natural Language Processing via Large Pre-Trained Language Models: A Survey. ArXiv (Nov 2021)
18. Modipa, T.I., Davel, M.H.: Two Sepedi-English code-switched speech corpora. Language Resources and Evaluation **56**(3), 703–727 (Sep 2022)
19. Modipa, T.I., De Wet, F., Davel, M.H.: Implications of Sepedi/English code switching for ASR systems. Pattern recognition association of South Africa (PRASA) (2013), accepted: 2014-11-03T13:17:24Z
20. Moila, M.M., Modipa, T.I.: The development of a sepedi text generation model using long-short term memory. In: Proceedings of the 2nd International Conference on Intelligent and Innovative Computing Applications. pp. 1–5. ACM, Plaine Magnien Mauritius (Sep 2020)
21. Nguyen, D., Dogruoz, A.S.: Word level language identification in online multilingual communication. In: Proceedings of the 2013 conference on empirical methods in natural language processing. p. 857 862 (Oct 2013)
22. Nguyen, L., Bryant, C., Kidwai, S., Biberauer, T.: Automatic language identification in code-switched Hindi-English social media text. Journal of Open Humanities Data **7**, 7 (Jun 2021)
23. Poplack, S.: Sometimes I'll start a sentence in Spanish Y TERMINO EN ESPAÑOL: Toward a typology of code-switching. Linguistics **51**(s1), 11–14 (Aug 2013)
24. Pratapa, A., Bhat, G., Choudhury, M., Sitaram, S., Dandapat, S., Bali, K.: Language modeling for code-mixing: the role of linguistic theory based synthetic data. In: Proceedings of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1543–1553. Melbourne, Australia (Jul 2018)
25. Puttkammer, M., Schlemmer, M., Pienaar, W., Bekker, R.: NCHLT Sepedi text corpora (May 2014)
26. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language Models are Unsupervised Multitask Learners. openAI blog **1**(8), 9 (Feb 2019)

27. Ramalepe, S., Modipa, T.I., Davel, M.H.: The analysis of the Sepedi-English code-switched Radio news corpus. Journal of the Digital Humanities Association of Southern Africa **4**(01) (2022), number: 01
28. Ramalepe, S.P., Modipa, T.I., Davel, M.H.: The development of a Sepedi text generation model using transformers. In: Proceedings of South Africa Telecommunication Networks and Applications Conference (SATNAC). pp. 51–56. Fancourt, Western Cape, South Africa, (2022)
29. Samanta, B., Reddy, S., Jagirdar, H., Ganguly, N., Chakrabarti, S.: A deep generative model for code switched Text. Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence pp. 5175–5181 (Aug 2019)
30. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, , Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017)
31. van der Westhuizen, E., Thomas, N.: A first South African corpus of multilingual code-switched soap opera speech. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (2018)
32. Yenduri, G., M, R., G, C.S., Y, S., Srivastava, G., Maddikunta, P.K.R., G, D.R., Jhaveri, R.H., B, P., Wang, W., Vasilakos, A.V., Gadekallu, T.R.: Generative Pretrained Transformer: A Comprehensive Review on Enabling Technologies, Potential Applications, Emerging Challenges, and Future Directions (May 2023)
33. Zhu, C., Ping, W., Xiao, C., Shoeybi, M., Goldstein, T., Anandkumar, A., Catanzaro, B.: Long-Short Transformer: Efficient Transformers for language and vision. In: Advances in Neural Information Processing Systems. vol. 34, pp. 17723–17736. Curran Associates, Inc. (2021)