



UNIVERSITY OF NAIROBI

FACULTY OF SCIENCE AND TECHNOLOGY

DEPARTMENT OF COMPUTING AND INFORMATICS

CCI 599: PROJECT REPORT

**English – Bukusu Automatic Machine Translation for Digital
Services Inclusion in e-Governance**

NGONI VELMA NAMWELA

P52/38259/2020

SUPERVISOR

DR LAWRENCE MUCHEMI

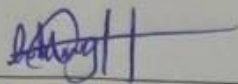
Project documentation submitted in partial fulfillment of the requirement for the award of

Master of Science in Computational Intelligence of the University of Nairobi

JULY 2022

DECLARATION

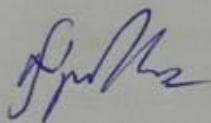
This project is my original work and to the best of my knowledge this work has not been submitted for any other award in any University.



Ngoni Velma Namwela

Date: 14th September 2022

This project report has been submitted in partial fulfillment of the requirements of the Master of Science in Computational Intelligence of the University of Nairobi with my approval as the University mentor.



Dr Lawrence G Muchemi

Department of Computing and Informatics

Date: 16/09/2022

TABLE OF CONTENT

DECLARATION	ii
TABLE OF CONTENT	iii
TABLE OF FIGURES	v
TABLE OF TABLES	vi
ABSTRACT.....	vii
1 INTRODUCTION	1
1.1 Background Information.....	1
1.2 Problem Statement.....	2
1.3 Research Objectives.....	3
1.4 Significance of the Research.....	3
1.5 Assumptions and Scope of the Research	4
2 LITERATURE REVIEW	5
2.1 Definition of Machine Translation.....	5
2.2 Machine Translation Approaches	5
2.3 Issues and Challenges faced with the different Machine Translation Approaches.....	9
2.4 Natural Language Processing for Low-resourced Languages	10
2.5 Existing Translation Models for Low-Resourced Languages.....	11
2.6 Evaluation of Machine Translation.....	13
2.7 Existing Machine Translation Tools.....	14
2.8 Other Authors Related Work and Findings.....	17
2.9 Research Gap	19
2.10 Conceptual Framework.....	19
3 RESEARCH DESIGN AND METHODOLOGY	21
3.1 Introduction.....	21
3.2 Quantitative approach	21
3.3 Research Data	21
3.4 Exploration (Modelling)	22
3.5 Prototype Design.....	23
3.6 Prototype Development.....	25
3.7 Evaluation (Testing Methodology).....	27
3.8 Ethical Considerations	27
4 RESULTS AND DISCUSSIONS.....	29
4.1 Exploratory Data Analysis Results	29

4.2	Model Performance Results	29
4.3	Prototype Evaluation Results	30
4.4	Discussions	31
5	CONCLUSION.....	34
5.1	Discussion	34
5.2	Research Contribution.....	35
5.3	Limitations in Research	36
5.4	Recommendation for Future Work	36
	REFERENCES	37

TABLE OF FIGURES

Figure 1: Figure showing Rule-based Machine Translation	6
Figure 2: Figure showing the architecture of Recurrent Neural Networks	7
Figure 3: Figure showing Bidirectional Neural Network	8
Figure 4: Figure showing a simple architecture of an Encoder-Decoder Model	9
Figure 5: Figure showing Google Translate System.....	14
Figure 6: Figure showing the Kamusi Project Homepage	15
Figure 7: Figure showing Unbabel's Integration.....	16
Figure 8: Figure showing IBM Watson Language Translator Demo.....	16
Figure 9: Figure showing Microsoft Translator for Bing.....	17
Figure 10: Figure showing the conceptual design of the research.	20
Figure 11: Figure showing the Exploratory Phase of the Research	22
Figure 12: Figure showing a mockup of the expected interface.	24
Figure 13: Figure showing the architecture of the prototype.....	25
Figure 14: Figure showing the data preprocessing done on the Jupyter Notebook	25
Figure 15: Figure showing the implementation of Embedded RNN of the Translation system	26
Figure 16: Figure showing the translation page of the prototype	26
Figure 17: Figure showing the evaluation page of the prototype.....	27
Figure 18: Figure showing the metrics of the Training Phase of the Encoder-Decoder model	32
Figure 19: Figure showing the metrics of the Training Phase of the Bidirectional RNN model	32

TABLE OF TABLES

Table 1: BLEU and NIST scores for Bidirectional Machine Translation Task.....	17
Table 2: Table showing the requirements for prototype design and implementation.....	24
Table 3: Figure showing the summary of the research data.....	29
Table 4: Table showing the evaluation scores for the first run of the research Exploration Phase.....	30
Table 5: Table showing the evaluation scores for the final run of the research Exploration phase.....	30
Table 6: Table showing sample translation results of the evaluation set.....	31
Table 7: Table showing the BLEU score comparison of different translations.....	33

ABSTRACT

It is important for human beings to communicate globally, and due to difference in languages, a translator is vital for effective use of digital services. The international marketers for example, use about ten languages, Kiswahili excluded, despite it being a national language in most of the Sub-Saharan African countries. According to The Cambridge Encyclopedia of the English Language, an estimate of 9% of Kenyans speak in English which is a major international language. The other 91% of Kenyans who don't speak in English either speak in Swahili or their tribal languages hence are excluded digitally. Even though the vernacular languages are spoken by approximately fifty million people in Kenya, they are resource-scarce from a language technological point. Machine translation models for these low-resourced African languages are scarce causing a lack of digital inclusion for many Africans. An English-Indigenous Language translator thus should be designed for digital inclusion of these non-English speaking communities. In this study, exploratory methodology was used to develop the English to Luhya machine translation prototype. Exploratory research is a methodology approach that investigates research questions that have not previously been studied in-depth. The model was successfully developed using Encoder – Decoder. It had a hidden layer size 128 and embeddings had 256 units. The training run for 50 epochs in batches of 100. The ADAM optimizer was used with a constant learning rate of 0.0005 to update the model weighs. The model was evaluated using BLEU score as the main evaluation metric and WER, SER, TER complementing the results. The model scored a highest BLEU score of 0.55, just 0.05 shy off the median range of 0.6 – 0.7 that has been achieved by other researchers. Compared to similar research on low-resourced languages, it scored modestly but outperformed translation of English to Kiswahili (0.20) using Statistical Machine Translation. For future work, the key initiative is creation of publicly available corpus which will serve as a catalyst to for research in this area. This limitation can benefit from having audio resources through speech recognition and speech-to-text implementation since Bukusu is primarily spoken and the lack of standardization in writing complicates the creation of clean reference sets and consistent evaluation. This study could also benefit from having reference models for Bukusu Named Entity Recognition and Parts of Speech tagging to improve translation accuracy. Since Bukusu language structure is like Swahili, key focus should first be in developing open-source NLP tools for Swahili language. With this, researchers Bukusu and other low resourced in other East Africa Bantu languages can be able to transfer the Swahili models and annotations to their respective languages.

1 INTRODUCTION

1.1 Background Information

The essential form of human communication in the Information Age is language, which acts as a transporter of information. Nevertheless, it has been viewed as an impediment to intercultural contact, particularly in urban and marginalized communities. Translating texts from one language into another quickly and accurately has become difficult (Sirbu, 2015).

With 1.2 billion people, Africa is both the second largest and second-most-populous continent in the world. With between 1500 and 2000 different African languages, it has a diverse language population (Doochin, What Are The Languages Spoken In Africa?, 2019). Due to commerce and intermarriage between various linguistic groups, the continent has a lengthy multilingual past. The Afroasiatic, Nilo-Saharan, Niger-Congo, Khoe, Austronesian, and Indo-European language families all include African languages. Following their independence, many African nations made their colonizers' language their official tongue for use in business, government, and education. However, most African nations continue to support multilingualism through local language promotion and appreciation programs.

About 40 major ethnic groups make up our multicultural nation of Kenya. With so many different languages spoken inside its boundaries, the nation is multilingual due to its unique ethnic makeup. Swahili and English are the two officially recognized legal languages among these (Sawe, 2017). Large corporations, universities, and the government are the main English-speaking environments. For instance, most of the legislation submitted to the National Assembly is written in English. Due to its broad use in trade, commerce, communications, and education, Swahili is regarded as the lingua franca of southeastern Africa. Kiswahili is almost primarily used in small-scale trade, the media, and educational institutions, with significant ties to urban life and certain vocations (Doochin, What Languages Are Spoken In Kenya?, 2019). Nearly 50 indigenous languages, including Kikuyu, Luhya, Kamba, Somali, Dholuo, Kalenjin, Arabic, Hindustani, and Punjabi, are spoken as Kenya's vernaculars.

Overcoming the language barrier has become a widespread issue in the global community due to the web's rapid progress and the integration of the world economy. With the democratization of ICT, there is a need for inclusion in providing information, work, and leisure opportunities on the internet. As the web becomes continuously entrenched in the lives of individuals, communities, and commerce, it is more vital than ever to ensure digital literacy for everybody and bridge this digital (Sanders, 2020). Language barriers are the most common reason an individual would not be an internet user, especially in rural

towns. The need for localization of internet products cannot be met by human translation, hence the use of machine translation to assist users in finding information has become a fundamental trend. Machine translation is the process of using a computer to translate a natural source language into a different natural target language. (Peng, 2013).

African languages that are frequently spoken have been the subject of machine translation efforts. However, most African languages are regarded as low-resource due to the difficulty in obtaining data, the lack of sufficient labeled audio speech, or the lack of concurrent translation across the various languages. (Cracking the Language Barrier for a Multilingual Africa, 2021). Most translation research for African languages has been done by the Masakhane project – open-source Natural Language Processing research that is continent-wide, distributed and online – such as machine translation, text-to-speech, document classification, keyword spotting, and sentiment analysis datasets (Masakhane, 2021). Most machine translation efforts in Kenya have been made for Swahili, Kikuyu, and Dholuo through document translation, language interpreting, transcription, and subtitling solutions. These translations have connected businesses to the African market. In the globalized marketplace, translation services have played a vital role across all sectors, such as business, financial, medical, legal, and marketing, thus breaking communication barriers (African Translation Solutions by Translate 4 Africa, 2021).

In this information age, Machine Translation can translate internet content into local languages to facilitate social inclusion and help all people contribute to and benefit from the digital economy and society. (GovernmentOfKenya, 2014). This will significantly improve access to digital services for those who have difficult access: those in rural areas, the elderly, and users of minority languages.

1.2 Problem Statement

Human beings need to communicate globally, and due to differences in languages, a translator is vital for the effective use of digital services. International marketers, for example, use about ten languages, of which Kiswahili is excluded, despite being a national language in most Sub-Saharan African countries. According to The Cambridge Encyclopedia of the English Language, 9% of Kenyans speak English, a primary international language. The other 91% of Kenyans who do not speak English either speak in Swahili or their tribal languages and are excluded digitally. Even though approximately fifty million people speak the vernacular languages in Kenya, the languages are resource-scarce from a language technological point of view. Machine translation models for these low-resourced African languages are scarce, causing many Africans to lack digital inclusion. An English-Indigenous Language translator should thus be designed to include these non-English speaking communities digitally.

1.3 Research Objectives

1.3.1 Overall Objective

To build a machine translation model for translating English text to Luhya text.

1.3.2 Research Objectives

1. To investigate the machine translation techniques currently applied in translating low-resourced African languages.
2. To investigate the factors affecting the implementation of automatic machine translation of low-resourced African languages.
3. To collect and analyze data for Natural Language Processing in the automatic machine translation model.
4. To develop and validate a model for automatic machine translation from English to Luhya.

1.3.3 Research Questions

1. What are the machine translation techniques being used for low-resourced African languages?
2. What are the factors affecting the implementation of machine translation for African languages?
3. What NLP tasks need to be performed on the language data to build the translation engine?
4. How can machine translation be implemented for English to Luhya translation?
5. What evaluation techniques should be applied to examine the performance of the English to Luhya translation engine?

1.4 Significance of the Research

With global digitization, products and services are getting to the market faster. On the other hand, learning different languages cannot keep up with this pace. As such, it is far easier to label products in the target market's language than to teach the entire market region how to speak a new language whenever a new product launches. Using local languages means that the users of a product can then relate better to the products as it makes them feel that they have been adequately considered. This project is aimed at helping in improving localization and internationalization works, and the focus is on Luhya since most popular machine translation engines have focused on European languages and left the African languages relatively underrepresented (Okpor, 2014).

Machine translation of local languages serves as a testbed for developing NLP technologies that perform reasonably well despite the low-resource constraint. By creating guidelines and providing training through open educational resources in collaboration with national institutions, this improves the capacity for the development of open language datasets and language technology applications and raises the

number of digitally accessible vernacular projects that other researchers can use in corpus-based research in African Language Technology. (Cracking the Language Barrier for a Multilingual Africa, 2021).

Research and development in Machine Translation to a local Kenyan language enables the digital inclusion of marginalized communities, in line with Kenya's Digital Blueprint towards achieving Vision 2030 under the Social and Economic Pillar (GovernmentOfKenya, 2014).

In addition to this, it is the requirement by the Communication Authority of Kenya that at least 60% of the content that Kenyan media companies (television and radio) must be local (Regulation By The Communications Authority of Kenya, 2018). Implementation of this project will be in line with this directive.

1.5 Assumptions and Scope of the Research

1.5.1 Assumptions

All data to be used in training the model is available and accessible for use in open research. The data to be collected is not proprietary to the source site.

1.5.2 Scope

This research is limited to translation to the Bukusu dialect. All data is textual, and none will be sourced directly from the population. The data collected will not be limited to a specific topic to ensure a good quantity of data and improve translation accuracy. The translation model built will then be applied to translate Government documents to Luhya text.

2 LITERATURE REVIEW

2.1 Definition of Machine Translation

'Machine translation is an area of research where computational linguists try to search for ways of utilizing computers to translate content from one natural language to another natural language' (Bowker & Ciro, 2019). It is used in information access across the internet, aiding human translators and communication between people.

Natural languages are immensely complicated, making machine translation a valuable work. Many words have various meanings, sentences can be read in many ways, and specific grammatical structures in one language could not exist in another or not be translated easily. In addition, extra-linguistic elements like real-world knowledge have a role in effective translation.

A machine translation system develops an internal representation after first analyzing the input in the source language. This representation is changed and converted into a format appropriate for the intended language. The result is then produced in the intended language. Fundamentally, machine translation is just the replacement of words from one natural language with words from another. Despite this, a competent translation cannot be created just by recognition of full phrases and their closest analogues in the target language (Bowker & Ciro, 2019).

2.2 Machine Translation Approaches

2.2.1 Rule-based Machine Translation

Rule-based Machine Translation (RBMT) bases its translations on grammatical rules and performs grammatical analyses of the source and target languages. However, as RBMT heavily relies on lexicons, efficiency is only attained after a protracted length of time. The three phases of the RBMT process for applying linguistic rules are analysis, transfer, and creation. As a result, syntax analysis, syntax generation, and semantic creation are required for a rule-based system. (Sghaier & Zrigui, 2020)

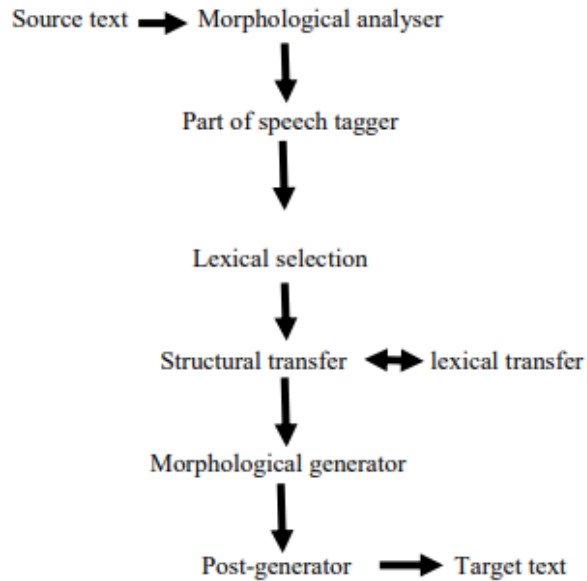


Figure 1: Figure showing Rule-based Machine Translation

2.2.2 Corpus-based Machine Translation Approach

Corpus-based machine translation (also known as data-driven machine translation) is an alternative approach to machine translation that addresses the rule-based machine translation knowledge acquisition problem. A bilingual parallel corpus is what the name "Corpus-Based Machine Translation" (CBMT) refers to while learning about fresh incoming translations. This method makes use of parallel corpora, which are massive collections of unprocessed data. Text and its translations are included in this raw data, and these corpora are used to learn about translation. The two further sub-approaches of the corpus-based approach are statistical machine translation and example-based machine translation (Irfan, 2017).

2.2.2.1 Statistical Machine Translation

SMT (Statistical Machine Translation) uses statistical models that are derived from the examination of substantial amounts of multilingual text. Its goal is to establish the similarity of a word in the source language to a word in the target language. This is very well illustrated by Google Translate. SMT is excellent for basic translation, but its biggest flaw is that it ignores context, which frequently results in incorrect translations (Okpor, 2014).

2.2.2.2 Example-based Machine Translation

The foundation of example-based MT is a look for similar instances of sentence pairs in the source and target languages. Because examples are taken from substantial databases of bilingual corpora, example-based MT falls within the category of corpus-based techniques. Given the source sentence, sentences

from the source side of the bilingual corpus with similar sub-sentential components are retrieved, and their translations to the target language are then used to construct the entire translation of the phrase.

2.2.3 Hybrid Machine Translation

The use of many machine translation techniques inside a single machine translation system distinguishes hybrid machine translation from other machine translation methodologies. The inability of any one technique to achieve a satisfying degree of accuracy is the driving force behind the development of hybrid machine translation systems. The accuracy of the translations has been significantly improved by numerous hybrid machine translation systems which combine statistical and rule-based methods.

2.2.4 Neural Networks Machine Translation

Neural Networks Natural Language Processing techniques are based on deep learning. For instance, they learn sequence-to-sequence transformations directly, doing away with the requirement for statistical machine translation's intermediate phases of word alignment and language modelling.

2.2.4.1 Recurrent Neural Networks

In a recurrent neural network (RNN), the output from the step before this one is used as the input for the current step. It uses comparable parameters for each input and builds the output by carrying out comparable operations on all the inputs or hidden layers. Text sequences can be inputs for recurrent neural networks, outputs for reversing text sequences, or both. The output and cell state from each time step become inputs the following time in the network's hidden layers' loop. With the help of this recurrence, which acts as a kind of memory, contextual data can pass through the network in such a way that it can be applied to the activities of the network at the present time step (Introduction to Recurrent Neural Network, 2018).

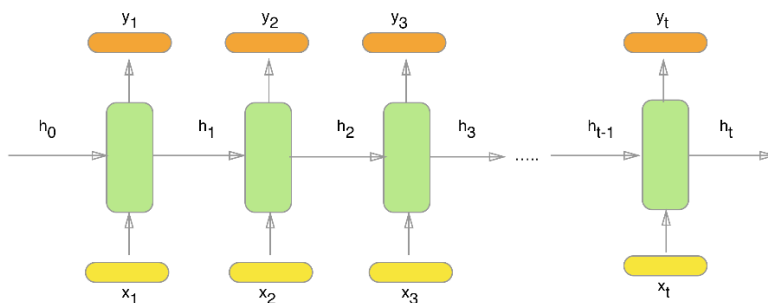


Figure 2: Figure showing the architecture of Recurrent Neural Networks

2.2.4.2 Bidirectional Neural Networks

A BRNN is made up of two RNNs, one of which starts at the beginning of the data sequence and moves forward, and the other of which starts at the end and moves backward. For one network, the input

sequence is fed in reverse time order; for another, it is fed in normal time order. At each time step, the outputs of the two networks are typically concatenated, however, there are other choices, such as summation. The networks may access both forward and backward information about the sequence thanks to this structure at each stage (Alla, 2022).

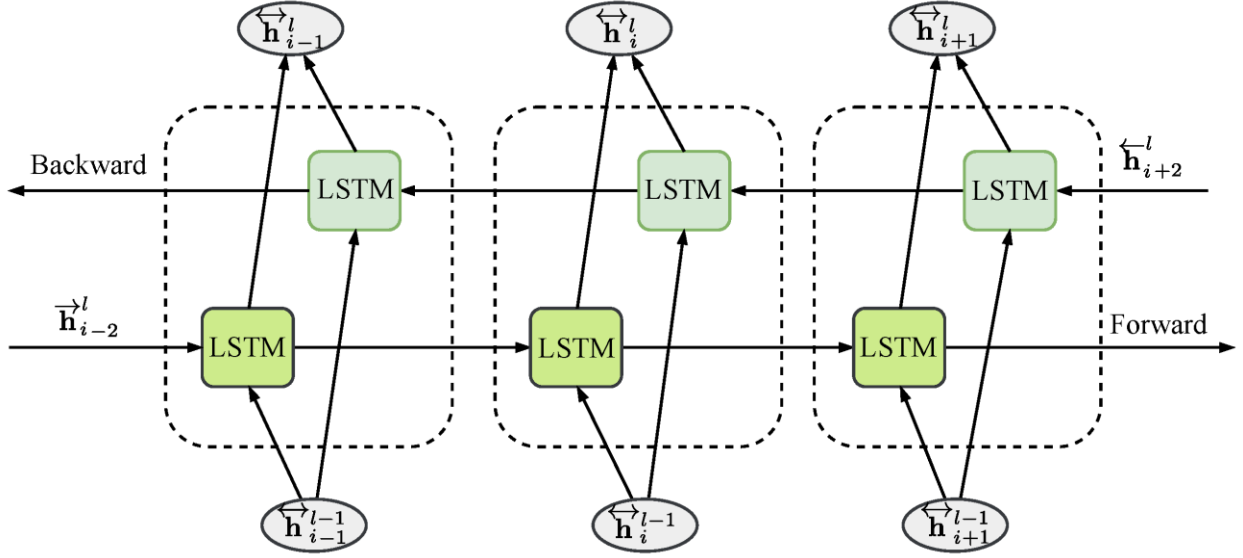


Figure 3: Figure showing Bidirectional Neural Network

To accommodate the backward training procedure, BRNNs have an additional hidden layer. The forward and backward hidden states are updated as follows for any given time t , where ϕ is the activation function, W is the weight matrix, and b is the bias.:

$$A_t(\text{Forward}) = \phi(X_t * W_{XA}^{\text{forward}} + A_{t-1}(\text{Forward}) * W_{AA}^{\text{forward}} + b_A^{\text{forward}})$$

$$A_t(\text{Backward}) = \phi(X_t * W_{XA}^{\text{backward}} + A_{t+1}(\text{Backward}) * W_{AA}^{\text{backward}} + b_A^{\text{backward}})$$

The hidden state at time t is given by a combination of $A_t(\text{Forward})$ and $A_t(\text{Backward})$. The output at any given hidden state is:

$$O_t = H_t * W_{AY} + b_Y$$

2.2.4.3 Encoder-Decoder Model

Recurrent neural networks are used in the encoder-decoder model to solve problems involving sequence-to-sequence prediction. When the input and output sizes vary, a sequence-to-sequence model attempts to map a fixed-length input with a fixed-length output. Two recurrent neural networks are used in the

method, one to encode the input sequence (referred to as the encoder) and the other to decode the encoded input sequence into the decoder target sequence. At each time step, the encoder takes one element from the input sequence, processes it, collects data on it, and propagates it forward. The model's encoder component created the final internal state, which is represented by the intermediate vector. It includes details about the whole input sequence to help the decoder provide precise predictions. Contrarily, the decoder considers the complete sentence and forecasts an output for each time step (Shreya Srivastava, 2019).

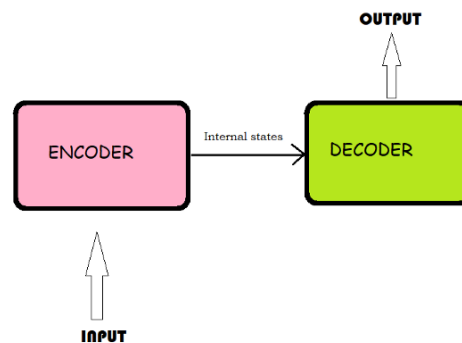


Figure 4: Figure showing a simple architecture of an Encoder-Decoder Model

2.3 Issues and Challenges faced with the different Machine Translation Approaches

In rule-based machine translation, there aren't enough high-quality dictionaries because creating new ones is expensive and certain linguistic data still needs to be manually entered. Dealing with colloquial language, ambiguity, and interactions between big-system rules is challenging. Although RBMT systems normally offer a way to discover new rules and expand and modify the lexicon, updates are typically quite expensive and frequently have unfavourable results.

In parallel corpora, a single sentence in one language may be translated into numerous sentences in the other, and vice versa. This is a problem with statistical machine translation. The Gale-Church alignment algorithm could be used to align sentences. Additionally, proper noun translations may be superseded with real-world training sets. When building a new statistical model (engine) to reflect a different vocabulary, dilution is another frequent oddity that results. SMT also has to contend with varying word ordering for various languages. For example, one can talk about SVO or VSO languages by referring to the conventional order of subject (S), verb (V), and object (O) in a phrase. There are also other

differences in word orders, for instance, where modifiers for nouns are located or where the exact words are used as a question or a statement.

To create the dependency trees required for the examples database and for analysing the phrase, Example-Based Machine Translation needs analysis and creation modules. Although parallel processing methodologies could be used, EBMT's computational efficiency is still a problem, especially for large databases.

2.4 Natural Language Processing for Low-resourced Languages

A low-resourced language is “a language that has fewer NLP resources, such that for a given task, there is no algorithm available to automatically do the task with adequate performance” (Duong, 2017). NLP for such low-resourced languages is important as it enables the preservation of the languages used in educational applications, knowledge expansion, monitoring of demographic and political processes and emergency response (Sciforce, 2019). The main NLP tasks applied to low-resourced languages are dependency parsing, POS tagging, cross-lingual word embedding and unwritten language processing.

2.4.1 POS Tagging

POS tagging assigns morphological categories such as nouns, adverbs and adjectives to tokens in a text. This task is performed in many NLP pipelines to help understand a language's syntax. It is mainly done using supervised learning algorithms to build a separate tagger for each target language and therefore relies on a large amount of data. However, in the case of low-resourced languages, data is scarce; therefore, unsupervised learning and semi-supervised algorithms are preferred. The unsupervised POS tagger does not require manually annotated data but instead groups words with the same morphosyntactic properties into clusters (Christodouloupoulos & Steedman, 2015). This presents two difficulties: first, letting the algorithm terminate the clusters on their own leads in clusters that are either too specific or too general, which is not ideal. To achieve good performance in tagging, semi-supervised POS tagging needs only a small amount of annotated data plus supervision from extra resources.

2.4.2 Dependency Parsing

The process of predicting the underlying structure of a sentence is known as parsing. Phrase-structure trees and dependency trees serve as the two main representations for achieving this. A delexicalized parser is used for languages with limited resources. It is created by taking away lexical features from a supervised parser before applying it to the target language. The basic assumption is that the two languages share characteristics outside just lexical items. The first delexicalized parser was developed by (Zeman, Univerzita, & Resnik, 2008), who used Danish, a closely related language, to build a parser for Swedish. For Swedish, they obtained a 66.4% F1 labelled attachment score. A delexicalized parser was also used

by (McDonald, et al., 2013). They experimented with 8 European languages and achieved a 78% F1 score.

2.4.3 Cross-lingual Word Embedding

Multiple lexicons from different languages are represented in the same dense vector space by cross-lingual word embeddings. They can cross-lingually represent the lexicons' syntactic and semantic features. Cross-lingual document classification, cross-lingual dependency parsing, and cross-lingual semantic parsing are frequently used in a transfer learning scenario (Zou, Richard, CER, & Manning, 2013). Many multilingual apps have also effectively used cross-lingual word embeddings. built on a multilingual dependency parser that can parse numerous languages by making use of numerous shared features across languages (Ammar, Mulcaire, Ballesteros, Dyer, & Smith, 2016). By using cross-lingual embeddings (Duong, 2017) created a semantic parser that can handle code-switching between English and German.

2.5 Existing Translation Models for Low-Resourced Languages

Below approaches are used in translating low-resource languages

2.5.1 Traditional Approaches

The first step in the process is data gathering, which compiles text or audio in the target language or languages. Typically, it produces a machine translation or POS engine or another NLP tool. These methods produce positive results but necessitate time-consuming data collecting and processing that frequently calls for professional assistance. The key disadvantage is that fresh corpora are needed for every new language introduced because the results are not directly transferable to other languages. Meanwhile, building up corpora for languages with limited resources is a crucial task. The Human Language Project (Abney & Bird, 2010) which outlines a common framework for annotated text corpora, is another example of the many languages approach.

2.5.2 Unsupervised Learning

Manually labelled data are not necessary for unsupervised learning. It includes unsupervised feature induction techniques such unsupervised POS tagging, Brown clustering, Word vector approaches, and unsupervised dependency parsing. Brown clustering is the process of organizing a vocabulary into word classes to create lexical representations based on the hunch that words with similar meanings will have words with similar distributions to their left and right. The foundation of many NLP strategies is word embeddings or word vectors. The co-occurrence matrix's zero entries, which formerly required large datasets, are now known to provide crucial information. A Positive-Unlabeled Learning approach was

developed by (Jiang, Yu, Hsieh, & Chang, 2018) to factorize the co-occurrence matrix and evaluate the suggested ways in four languages.

2.5.3 Transfer Learning

In transfer learning, certain similarities between languages may be taken advantage of to create, for instance, a language model for one language from a model for another. Cross-lingual transfer learning is the process of moving resources and models from sources with abundant resources to target languages with limited resources.

2.5.3.1 *Transfer of Annotation*

Building the POS tags, syntactic characteristics, and semantic features for such models using cross-lingual transfer learning typically requires linguistic knowledge and resources regarding the relationship between the source language and the destination language. Techniques that don't require auxiliary resources, such parallel corpora, are now available thanks to recent breakthroughs. According to (Kim, Kim, Sarikaya, & Fosler-Lussier, 2017) a cross-lingual model uses public BLSTMs for language-specific representations and private BLSTMs for knowledge transfer from other languages without utilizing linguistic knowledge between the source language and the destination language. To represent language-general information and maintain the information about a particular target language, the cross-lingual model is developed using language-adversarial training and bidirectional language modeling.

2.5.3.2 *Transfer of Models*

Transfer of models is the process of training a model in a language with abundant resources and then using it in a language with limited resources in zero-shot or one-shot learning. Zero-shot learning is the practice of training a model in one domain and expecting that it automatically generalizes to another area with fewer resources. A related method is known as "one-shot learning," which modifies a model developed in a domain with abundant resources using a small number of instances from a domain with scarce resources. The weights gathered for a language pair with rich resources are transferred to language pairs with limited resources in a method known as machine translation. An illustration of such a strategy is a model by (Zoph, Yuret, May, & Knight, 2016). A parent model (French to English) is trained, and some of the trained weights are reused as the initialization for a child model (a specific low-resource language pair), which is further trained (Hansa, Turkish and Uzbek into English). (Nguyen & Chiang, 2017) investigated a similar strategy where the parent language pair is also low resource but related to the kid language pair.

2.5.3.3 *Joint Multilingual or "Polyglot" Learning*

This model trains a single model on a variety of datasets in all languages to enable parameter sharing when possible. It translates data in all languages to a shared representation (such as phones or

multilingual word vectors). The use of 100 huge majorities of approved languages and cross-lingual sentence embeddings to train a cross-lingual Transformer language model is roughly akin to the approach being used in the present. The latter method learns joint multilingual sentence representations for 93 languages that are written in 28 scripts and belong to more than 30 different language families. The method enables learning a classifier on top of the resulting sentence embeddings using only English annotated data and transferring it to any of the 93 languages without any modification. It does this by using a single BiLSTM encoder with a shared BPE vocabulary for all languages, coupled with an auxiliary decoder and trained on parallel corpora.

2.6 Evaluation of Machine Translation

To evaluate the effectiveness of machine translation models, a black-box evaluation is carried out. This is concerned with the system's objective behaviour in response to a specified evaluation set, which is a collection of source-language sentences and the translations the translation system produced for each of those sentences into the target language. The evaluation's De facto metrics are BLEU, WER, SER, NIST, METEOR, and TES. As a result of comparing the MT output with reference translations and providing comparison scores, all these metrics require reference translations. These metrics might be used to evaluate the output of any number of systems swiftly and automatically where reference translations are provided.

2.6.1 BLEU

BLEU is an IBM-created automatic metric. Unigram (single word) and high-order n-gram overlap between MT output and reference translations is measured using BLEU. It is defined as follows:

$$\text{BLEU} = \min(1, \frac{\text{length}}{\text{length}_{\text{ref}}}) \left(\prod_{i=1}^n \text{precision}_i \right)^{1/4} \cdot \text{E6}$$

Precision, or n-gram precision, is the primary element of BLEU. It is figured out as the proportion of matched n-grams to all n-grams in the evaluated translation. Each n-gram order's precision is determined independently, and the precisions are then averaged geometrically. The most popular definition of the highest n-gram order is four (four words in a sequence). Indirectly, the degree of grammatical correctness of a translation is assessed using higher-order n-grams. The shortness penalty, which penalizes phrases that are shorter than the reference, is used in the BLEU metric to calculate the modified precision score.

2.6.2 Word Error Rate (WER)

WER is the proportion of words that must be added, subtracted, or substituted in the translation to produce the reference phrase. WER could be calculated automatically by taking into account the spacing between the two sentences. This metric's computation is quick and accurate. The main disadvantage,

however, is that it depends on the reference sentences. For an identical sentence, there are virtually infinitely many accurate translations; nevertheless, this metric only accepts one as correct.

2.6.3 Sentence Error Rate (SER)

SER displays the proportion of sentences where the translations and references do not agree. It exhibits the same benefits and weaknesses as WER. There are some variations on WER that have been defined and that can also be obtained automatically. Multi reference WER (mWER): This variation uses a similar approach to WER but considers multiple references for each sentence that needs to be translated. For each sentence, the editing distance will be calculated among the various references, and the smallest one will be chosen. Its disadvantage is that it requires a lot of human effort before it can be used.

2.6.4 Translation Edit Rate (TER)

The WER is a derivate of the TER metric. It makes use of a further editing procedure called shifts of word sequences (Shift). A shift relocates a continuous string of words from one place in the assessed translation to another. If more than one reference is available, only the number of edits to the closest reference is counted because only the bare minimum of edits is required to change the translation. TER is normalized using the reference's mean length. The position-independent error rate (PER), which regards the reference and translation output as bags of words, is derived from the word error rate (WER). Without regard for location, words from the translation are aligned to those from the reference.

2.7 Existing Machine Translation Tools

2.7.1 Google Translate

Using Google Translate, users may translate text, documents, and webpages across several languages using neural machine translation technology. Google Translate supported 109 languages at various levels as of August 2021, and as of April 2016, it claimed to have over 500 million users worldwide, translating more than 100 billion words every day. (Wikipedia, 2021).

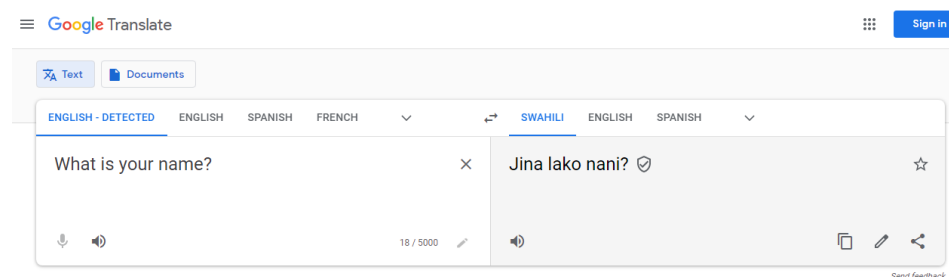


Figure 5: Figure showing Google Translate System

2.7.2 The Kamusi Project

A collaborative online dictionary called The Kamusi Project seeks to create dictionaries and other linguistic materials for every language and make them available to everyone. Users can add material and register. Since 2010, additional languages have been incorporated into programming and the Swahili-English database. All existing languages can have connected dictionaries built as part of the Kamusi project. Excellent materials for one-word translation can be found at the Kamusi Project. However, it was not intended to be a phrase translator when context extraction is required to determine the meaning of words and phrases in context.

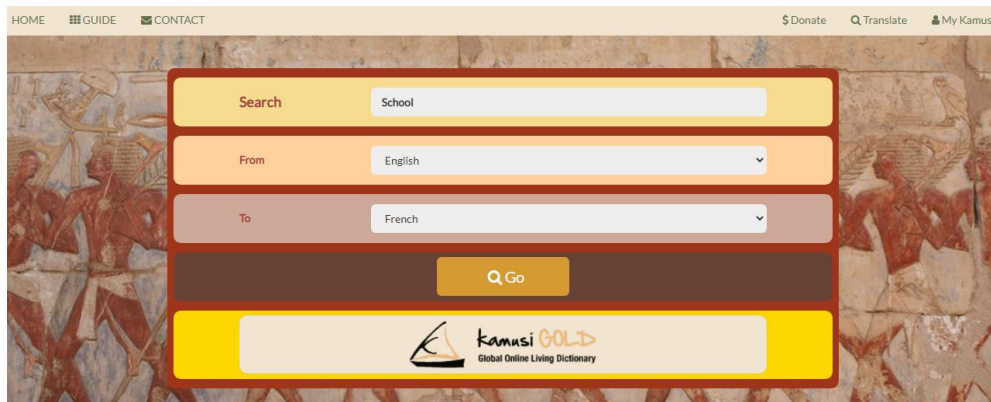


Figure 6: Figure showing the Kamusi Project Homepage

2.7.3 Unbabel

By removing linguistic barriers, Unbabel enables commerce to flourish across borders and cultures. The organization's language operations platform combines sophisticated artificial intelligence with human editors to produce quick, effective translations of the highest calibre that get smarter over time. Unbabel enables agents to provide consistent multilingual help from their existing workflows and integrates smoothly across all channels, making it simple for businesses to expand into new markets and win over customers everywhere. To effortlessly distribute translations within current workflows across digital support channels like chat, email, or FAQs, Unbabel integrates with the most common CRMs and Chat platforms. All of this is controlled through the Portal, where users can manage translation workflows, keep an eye on vital statistics like speed or quality, and carry out additional operations to operationalize the use of several languages throughout their enterprise.



Figure 7: Figure showing Unbabel's Integration

2.7.4 IBM Watson Language Translator

Users can translate texts and documents from one language to another while maintaining formatting by using IBM Watson Language Translator. Users experience better translation accuracy and faster speeds when using the most recent Neural Machine Translation algorithms. The Watson Language Translator also enables the creation of unique, industry- or region-specific models for business requirements.

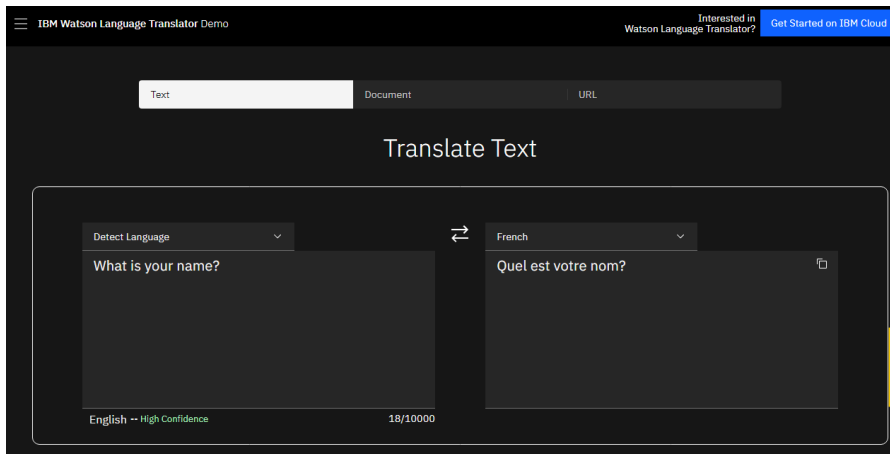


Figure 8: Figure showing IBM Watson Language Translator Demo

2.7.5 Microsoft Translator

This is a multilingual machine translation cloud service offered by Microsoft. Multiple consumer, development, and enterprise products all incorporate it. Additionally, it offers organizations cloud-based translation services for both text and speech. A free tier of the Translator Text API's text translation service supports two million characters per month, whereas subscription levels support billions. Based on the length of the audio stream, Microsoft Speech services provide speech translation.

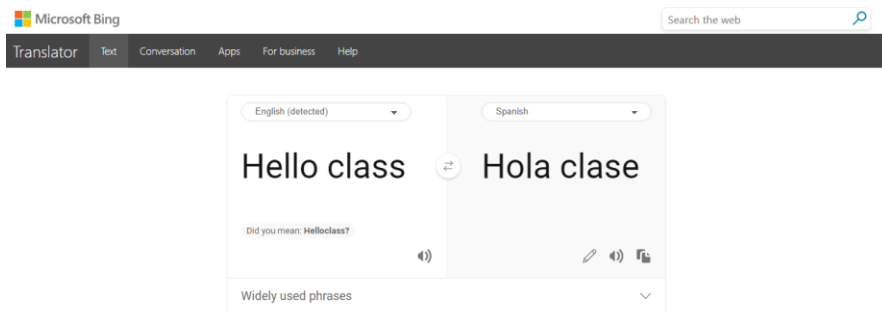


Figure 9: Figure showing Microsoft Translator for Bing

2.8 Other Authors Related Work and Findings

(Pauw, Wagacha, & Schryver, 2011) built a machine translation system using the SAWA corpus and the common MOSES package. Using the industry-accepted machine translation assessment metrics BLEU and NIST, the SMT system was developed on a 90% training set and tested on a 10% test set. They contrasted their findings with those of Google Translate's Swahili app. The SAWA technology underperforms when compared to Google Translate for translating from English to Swahili. Their approach performed better and was not hindered by the morphological generation problems of the target language for Swahili to English translation.

		BLEU	NIST
GOOGLE	English → Swahili	0.26	3.96
SAWA	English → Swahili	0.20	2.92
GOOGLE	Swahili → English	0.29	4.14
SAWA	Swahili → English	0.35	4.52

Table 1: BLEU and NIST scores for Bidirectional Machine Translation Task

An open-source toolkit for statistical machine translation is described by (Koehn, et al., 2007) its contributions support linguistically motivating variables, confusion network decoding, and effective data formats for translation and language models. The toolkit also contains a wide range of tools for training, adjusting, and applying the system to other translation jobs in addition to the SMT decoder. The toolkit is an all-inclusive, pre-configured translation system for scholarly study. It includes all the elements required to prepare the data for preprocessing and train the language and translation models. Additionally,

it includes instruments for fine-tuning these models using minimum error rate training and assessing the translated text using the BLEU score. The MT research community will benefit from the open-source tools that have been presented in their study. In a clear and adaptable framework, they described a novel SMT decoder that can add some language aspects. Many new possibilities and problems are raised by this new line of inquiry, necessitating more study and testing. Initial results illustrate the potential usefulness of variables for statistical machine translation.

The Anusaaraka system, which translates from English to Hindi, adheres to the fundamentals of information preservation (Bharati, 2010) created a system that allows access to texts written in one Indian language in another. To analyze the provided English text, it makes use of an XTAG-based super tagger and a light dependency analyzer created at the University of Pennsylvania. It creates new ways to divide the workload between people and machines. With respect to a specific input, the system generates a number of outputs. The computer handling the lexical load and leaving the human with the syntax load results in the output that is as simple as it can be. A comprehensive parser and a multilingual dictionary are used to produce output that is based on the most thorough examination of the English input text.

(Zoph, Yuret, May, & Knight, 2016) share their work on a case study on different machine translation techniques for translating between the languages of Malayalam and English. Finding the benefits and drawbacks of various strategies is one of the study's goals. They describe the development of English to Malayalam and Malayalam to English baseline phrase-based SMT systems, and the evaluation of its performance compared against the RBMT systems, using comparisons such as Statistical MT (SMT) vs RuleBasedMT (RBMT), English to Malayalam SMT vs Malayalam to English SMT, and English to Malayalam RBMT vs Malayalam to English RBMT. They conclude that SMT systems perform better than RBMT systems, English-Malayalam systems perform better than Malayalam-English systems, and Malayalam-to-English systems perform better than English-to-Malayalam systems in the case of RBMT. They outline the conditions for integrating morphological processing into the SMT in order to increase translation accuracy based on their assessments and thorough error analyses.

In their study Neural machine translation for low-resource languages, (Ostling & Tiedemann, 2018) researchers take a token-by-token approach to the source sentence, creating one (potentially empty) chunk of the target sentence at a time. Following that, the created chunk is introduced into the partially target text in a position foreseen by a reordering process. Since the data is not incredibly sparse at the token level, they translate each token using a character-to-character model dependent on the local source context. Additionally, the source and destination languages' open vocabularies are a result of this. They use the EFMARAL aligner to train the reordering model, which is supervised by word alignments.

Their model requires a collection of target tokens and a set of source tokens that are both the same length. We do this by first identifying the most certain word alignments. The final training sequence has the same length as the original sentence since they use it as a fixed point. It is presumed that unaligned source tokens produce an empty string. It is assumed that the entire sequence is produced by source tokens that are aligned to a target token, followed by unaligned target tokens (with spaces between tokens). Following the extraction of the aligned token sequences, backpropagation with stochastic gradient descent is used to train their model. With BLEU scores of 9–17% (in-domain), their model bridges the gap between phrase-based and neural machine translation with only roughly 70 000 training data tokens. In this situation, the conventional NMT system is unable to generate any logical output.

2.9 Research Gap

In less than ten years, neural machine translation has had an enormous development and has already reached a mature stage. Nonetheless, the method still performs poorly on language pairs with limited resources compared to language pairs with high resources because there aren't any vast parallel corpora available. Therefore, it is essential to develop low-resource language translation systems, and this area of research has gained popularity in neural machine translation. In light of this, semi-supervised techniques have been used to translate low-resource languages, creating pseudo-parallel data from monolingual data and then using artificial data to train neural machine translation systems. (Shi, Wu, Su, & Huang, 2022). Unsupervised learning is another trend, when researchers begin to build systems fully reliant on monolingual corpora with the aid of certain deep learning advancements. Researchers create many unsupervised models from various angles; for instance, some techniques concentrate on the modification of model structure, while others are built on the transfer of language information between different languages (Artetxe, Labaka, & Agirre, 2017). Other trends include Transfer Learning, Word embeddings, Data Augmentation, Pivot-based methods, and Syntax Enhanced methods.

This research aims to train a model to translate English to Luhya (Bukusu dialect) using modern neural networks machine translation techniques. The model will be trained using different algorithms to identify the optimal algorithm for translating English to Bukusu. Unsupervised learning through word embeddings will be used to build the translation model. Evaluation will then be done using automatic metrics to compare the translation accuracy and later evaluated by Human Expert Evaluators.

2.10 Conceptual Framework

The steps that must be taken in the study to accomplish the research objectives are represented by a conceptual framework. The conceptual framework for the study, which was developed following a literature assessment, is shown below.

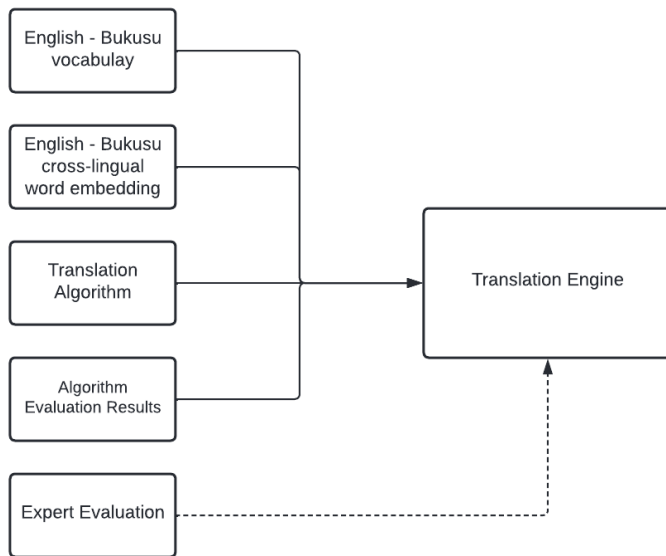


Figure 10: Figure showing the conceptual design of the research.

The model is suitable for this study due to the nature of the research data and the limited resource available. The model supports online secondary data, as other models would require fieldwork to collect primary data.

3 RESEARCH DESIGN AND METHODOLOGY

3.1 Introduction

The chapter covers the research design, the research data and population, the data collection measures and tools used, the data preprocessing methods used to prepare and clean the data for translation, and the proposed translation models.

3.2 Quantitative approach

Research design is a blueprint for collecting and analysing data created to answer questions guiding the research. It articulates the required data, the methods used to gather it, and how the research questions will be answered by the data collected. It also evaluates the research purpose, methods and approaches and time limit of a research study.

The prototype of the English to Luhya machine translation was created using exploratory methods. Exploratory research is a methodology that looks at research problems that haven't been thoroughly investigated before (George, 2021).

The Exploratory research design was selected for this study because:

1. There is limited information from past research on machine translation models for English to Bukusu. This meant the researcher spent time studying materials concerning Natural Language Processing and translation of low-resourced African languages.
2. Two research goals of this study were to investigate the factors affecting the implementation of machine translation of low-resource African languages and investigate the techniques currently applied in translating low-resource African languages—the results of these research questions aided in acquiring more information about the research study.

3.3 Research Data

Data used in this research was English and Luhya text data. Numerical data was not gathered as the research area was Human Language Technologies. Five thousand records of parallel text in English and their Luhya translation were collected. This was done through the Primary approach in which the researcher extensively collected the data from different sources as no English-Bukusu dataset is publicly available for Natural Language Processing research. The data was collected from English – Bukusu Bible translation, English – Bukusu dictionary and crawling of sites, i.e., Mulembe Nation, Globse and Lugha Yangu.

NLP data preprocessing was then performed on the data to clean it and prepare it into a usable form for building the model. The below tasks were performed on the dataset:

1. Removing HTML tags, non-ASCII symbols and punctuation marks from the text
2. Removing numbers that marked chapters in the hymn books and Bible
3. Converting the text to lower case
4. Removal of punctuation marks
5. Tokenising the text to break it into individual linguistic units.

3.4 Exploration (Modelling)

This phase of the research involved the development of a translation model. The process was repetitive, whereby the data was subjected to different Neural Network algorithms and evaluated each time against the metrics. This phase was stopped when results were found to be good enough to use in the prototype. This was done on Jupyter Notebook platform as it is suitable for data science tasks.

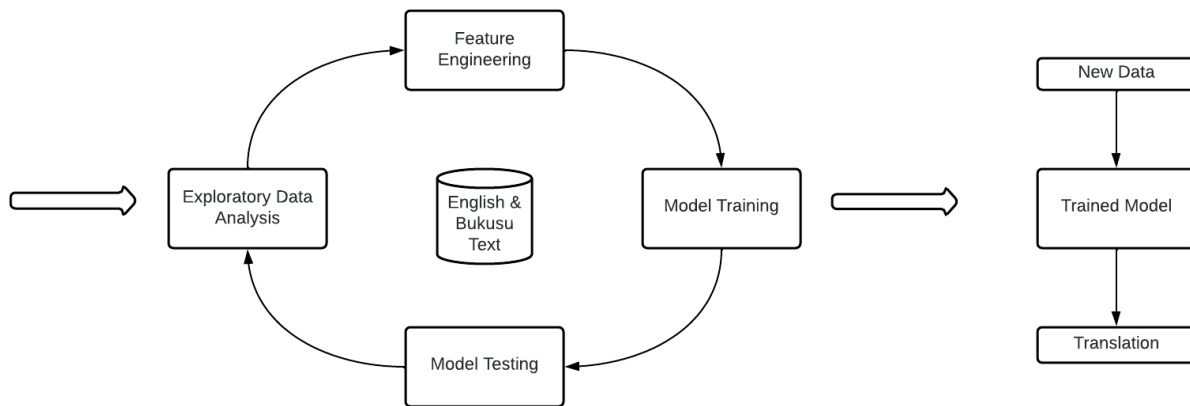


Figure 11: Figure showing the Exploratory Phase of the Research

3.4.1 Exploratory Data Analysis

To create insights and identify underlying hidden patterns in the data, exploratory data analysis was carried out to determine the basic properties of the text data. The tasks that were undertaken included exploring Ngrams and analyzing text statistics. Word frequency analysis, sentence length analysis, and average word length analysis were all part of the text statistics. Ngram exploration involved analysis of the unigrams, bigrams, and trigrams of the English and Bukusu text data. Exploration of the data provided an output of the language vocabulary. This informed the complexity of the translation problem as a more complex vocabulary requires intensive computing resources to learn a model.

3.4.2 Feature Engineering

Word embeddings were used to model the language. The algorithms that had been previously selected are neural networks, which work with numbers. Therefore, the words were converted to their corresponding numeric vector operations. Word embeddings also retain the relationship between words and store the relationship between subjects, verbs, and objects in the sentences. After establishing the word embeddings, the input embeddings of the translation model were padded. Padding is done because the input and corresponding output sentences can be of varying lengths.

3.4.3 Model Training

Deep Neural Networks algorithms were used to learn a mapping between the source space (English word vector embeddings) and the target (Bukusu word vector embeddings). Four algorithms were implemented: Simple RNN, RNN with embedding, Bidirectional Neural Networks and Encoder-Decoder model. An 80 percent training set and 20 percent testing set were created from the data. At each iteration of building the model, the model was subjected to the testing set, and the performance was reviewed against a set of metrics i.e., BLEU, WER, SER and TER. The model selected to build the system prototype had the highest BLEU score, low WER, low SER, and low TER.

3.5 Prototype Design

The prototype design aids in defining the system architecture as well as the hardware and system requirements. The following steps were taken in system design:

- 1. Machine Translation Engine Design:** This step involves designing a translation engine through performing data preparations and reviewing suitable implementation algorithms for low-resourced language translation.
- 2. User Interface Design:** This step will involve designing a user interface where users will be able to use the translation engine. The user interface will be presented as a Web Application as it is easy to build and is easily compatible with computational intelligence algorithms.

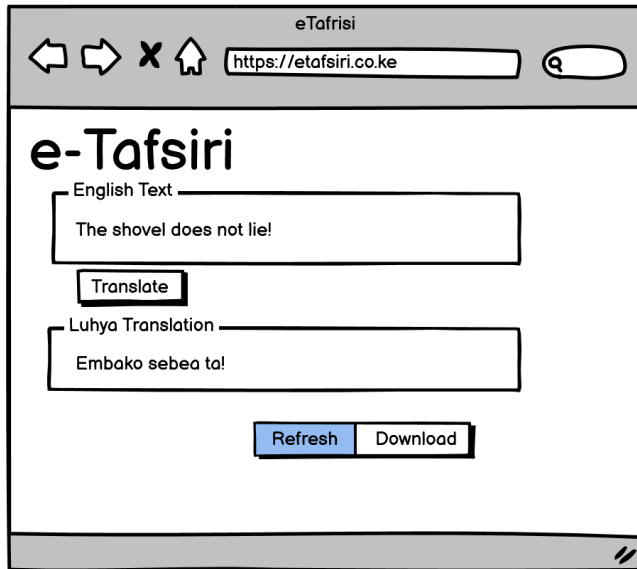


Figure 12: Figure showing a mockup of the expected interface.

3.5.1 Prototype Requirements

The below functional requirements will be considered when designing the user interface:

1. The system shall provide an interface for users to input their text for translation.
2. The system shall accept English input for translation to Bukusu.
3. The system shall provide an interface to display the Bukusu output text.

Below are the requirements needed to implement the translation prototype

Hardware	Software	Libraries & Algorithms	Data
<ul style="list-style-type: none"> • Intel® Core™ i5-7200U CPU @2.50 GHz 2.71 GHz • 8.00 GB RAM • x64-based Processor 	<ul style="list-style-type: none"> • Python 3.7 • Jupyter Notebook • Notepad++ 	<ul style="list-style-type: none"> • NLTK • Flask • Tensorflow 	<ul style="list-style-type: none"> • English text • Bukusu text

Table 2: Table showing the requirements for prototype design and implementation

3.5.2 Prototype Architecture

The proposed architecture of the research prototype follows the unsupervised learning approach of word vector embeddings. English text is input through an interface. The text is then preprocessed to normalise it for the NLP translation task. The text is then encoded numerically through word embeddings and

mapped to the corresponding Bukusu encoding. This is then decoded, and Bukusu output text is displayed.

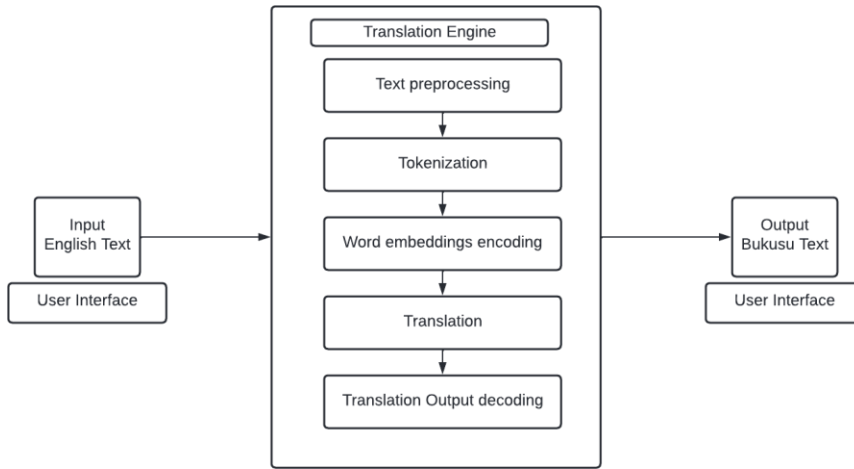


Figure 13: Figure showing the architecture of the prototype

3.6 Prototype Development

This section provides a high-level description of the tasks implemented to develop a prototype of the translation solution.

3.6.1 Data Preprocessing

The data preprocessing steps on the English and Bukusu text were done to normalise the data and prepare it for the NLP Machine Translation task.

```

def to_pairs(doc):
    lines = doc.strip().split('\n')
    pairs = [line.split('\t') for line in lines]
    return pairs

# clean a list of lines
def clean_pairs(lines):
    cleaned = list()
    # prepare regex for char filtering
    re_print = re.compile('[^%s]' % re.escape(string.printable))
    # prepare translation table for removing punctuation
    table = str.maketrans('', '', string.punctuation)
    for pair in lines:
        clean_pair = list()
        for line in pair:
            # normalize unicode characters
            line = normalize('NFD', line).encode('ascii', 'ignore')
            line = line.decode('UTF-8')
            # tokenize on white space
            line = line.split()
            # convert to lowercase
            line = [word.lower() for word in line]
            # remove punctuation from each token
            line = [word.translate(table) for word in line]
            # remove non-printable chars from each token
            line = [re_print.sub('', w) for w in line]
            # remove tokens with numbers in them
            line = [word for word in line if word.isalpha()]
            # store as string
            clean_pair.append(' '.join(line))
        cleaned.append(clean_pair)
    return cleaned
  
```

Figure 14: Figure showing the data preprocessing done on the Jupyter Notebook

3.6.2 Model Training and Testing

The translation model was trained and tested by splitting the data into a training and testing set. This data was then modelled using the four established Neural Networks algorithms and evaluated against the set evaluation metrics. This phase was done on Jupyter Notebook as it is compatible with the Machine Translation tasks.

```
Model Implementations and Evaluations

In [30]: def embed_model(src_vocab, tar_vocab, src_timesteps, tar_timesteps, n_units):
          model = Sequential()
          model.add(Embedding(src_vocab, n_units, input_length=src_timesteps, mask_zero=True))
          model.add(LSTM(n_units))
          model.add(RepeatVector(tar_timesteps))
          model.add(LSTM(n_units, return_sequences=True))
          model.add(TimeDistributed(Dense(tar_vocab, activation='softmax')))
          # Compiling the model
          model.compile(optimizer = 'adam', loss='sparse_categorical_crossentropy')
          model.compile(optimizer = 'adam', loss='categorical_crossentropy')
          # Summarising the model
          model.summary()
          return model

In [31]: embed_model = embedded_model(eng_vocab_size, bukusu_vocab_size, eng_Length, buk_Length, 256)

Model: "sequential"
-----
Layer (type)                Output Shape              Param #
-----
embedding (Embedding)       (None, 32, 256)          310016
lstm (LSTM)                  (None, 256)              525312
repeat_vector (RepeatVector) (None, 28, 256)          0
lstm_1 (LSTM)                (None, 28, 256)          525312
time_distributed (TimeDistr  (None, 28, 1618)         415826
ibuted)
-----
Total params: 1,776,466
```

Figure 15: Figure showing the implementation of Embedded RNN of the Translation system

3.6.3 User Interface Development

A web interface was developed to make the translation model usable in predicting new data translation. The web app was built using Flask framework as it permits easy interaction between the machine learning model and web interface. After completion, the application was hosted on the Google cloud.

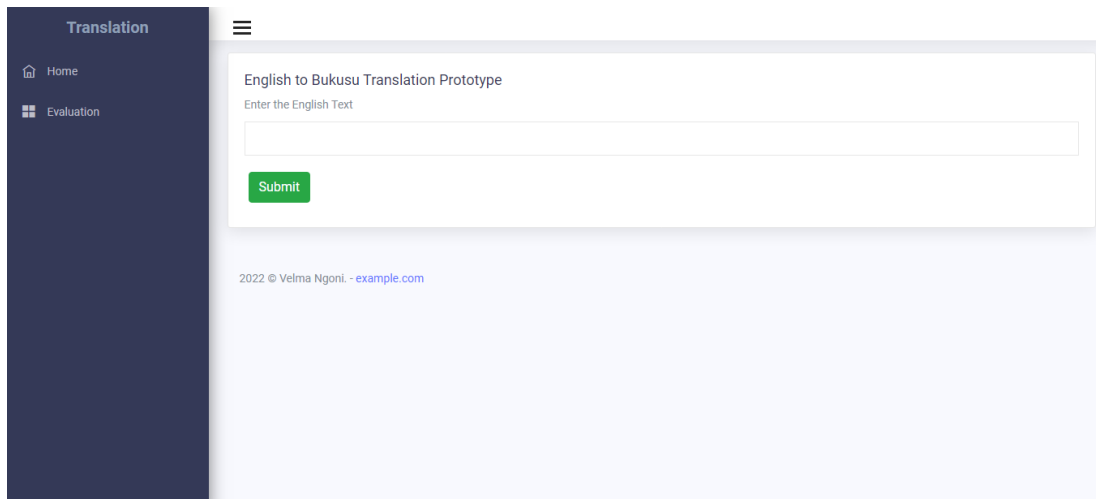


Figure 16: Figure showing the translation page of the prototype

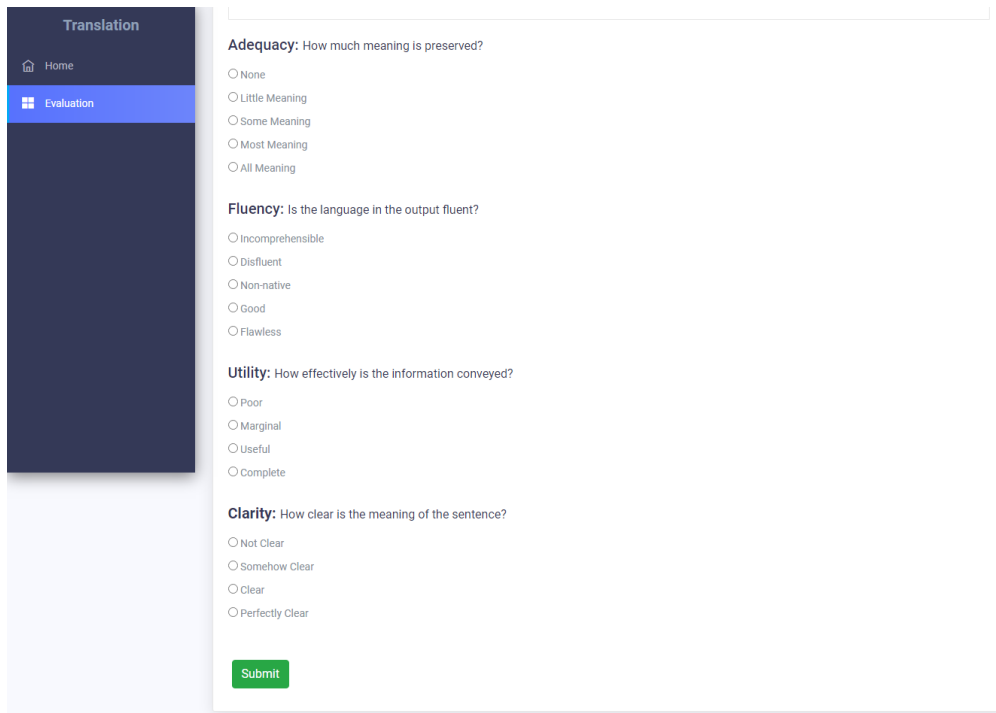


Figure 17: Figure showing the evaluation page of the prototype

3.7 Evaluation (Testing Methodology)

The strategies and testing types used to assess and certify that the application being tested satisfies the requirements are known as testing methodology (Hamilton, 2022). The methodology below was used to evaluate the prototype's performance after training and testing the different algorithms.

1. The researcher identified 20 independent sentences that were not in the dataset. These sentences were distributed to have 25% common phrases and 75% related to government services to show the performance in line with the research scope.
2. Human Experts identified from Trans Nzoia County provided the translations of these 20 sentences. This provided the baseline for the evaluation of the model.
3. The 20 sentences were input into the system and the output was recorded.
4. The system outputs were measured against the set metrics by capturing the BLEU score, Word Error Rate, Sentence Error Rate and Translation Edit Rate.

3.8 Ethical Considerations

1. **Cultural Sensitivity:** The system should have high translation quality so that the Luhya community and other communities are not offended by the results of the system. A word such as grandmother should be translated to 'Muloosi'; however, if translated to 'Omulosi', it means mad person.

2. Translation Accuracy: Translation accuracy should be considered in the development of the system. The intended use case of the system is e-governance. Therefore, correct information should be delivered to the users as they will access government information such as E-Citizen. The system should give accurate information for this.

4 RESULTS AND DISCUSSIONS

This chapter explains the performance outcome of the developed translation model when subjected to different testing scenarios. The model performance was evaluated on the translation accuracy, translation error rate and impact of changing the parameters when building the model.

4.1 Exploratory Data Analysis Results

Dataset	# Of Sentences	# Of English Tokens	# Of Bukusu Tokens
Training	4000	11240	14060
Testing	1000	8347	9337
Evaluation	20	132	167

Table 3: Figure showing the summary of the research data

One of the objectives was to collect and analyse NLP data for Machine translation. For the dataset collected, tasks performed were analysing the text statistics and Ngram exploration. For the English text, it was observed that most words averaged at 5 -7 characters with sentences ranging from 3 words to 12 words. On the Bukusu dataset, the words averaged at 9 -12 characters, with a sentence length ranging from 1 to 7 words.

From the word frequency, it was noted that the most frequent English words were articles and prepositions. It is also noted that 'God' frequently appeared in the text as most of the data were religious, which was similarly noted for Bukusu data.

English vocabulary had 11240 words while Bukusu vocabulary had 14060 words. This informed the complexity of the translation problem as 11240 was to be mapped to 14060 words. The words were then represented in numerical vectors. The data was then padded to have all data of the same length. The longest sentence in the input contained 12 words, and zeros were added in the empty indexes for sentences that contained less than 12 words.

4.2 Model Performance Results

The assessment of machine translation systems is a vital step in a research study. For this research, four algorithms were evaluated using automatic validation techniques. Below are the results achieved with different scenarios.

Algorithm	BLEU-1 SCORE	BLUE-2 SCORE	WER	SER	WIL
Simple RNN	0.06	0.01	0.97	0.98	0.98
Embedded RNN	0.06	0.01	0.99	0.98	0.98
Bidirectional RNN	0.11	0.04	0.94	0.93	0.94
Encoder-Decoder	0.13	0.05	0.91	0.91	0.89

Table 4: Table showing the evaluation scores for the first run of the research Exploration Phase

Unsatisfactory results were achieved with the first model training. It was noted that the data was little, and most sentences were unique. This called for the addition of data in which more data was collected for the model training.

Algorithm	BLEU-1 SCORE	BLUE-2 SCORE	WER	SER	WIL
Simple RNN	0.49	0.43	0.42	0.43	0.85
Embedded RNN	0.52	0.42	0.41	0.41	0.85
Bidirectional RNN	0.52	0.39	0.44	0.44	0.88
Encoder-Decoder	0.55	0.44	0.68	0.68	0.89

Table 5: Table showing the evaluation scores for the final run of the research Exploration phase.

4.3 Prototype Evaluation Results

Below are sample results for the 20 independent sentences evaluated using the translation system.

Sent 1	English	The government is committed to expanding health service coverage.
	Source	Buruki bwa Kenya khebunaya busilikhi khubulu mundu
	System	Gavumenti committed expanding byobulamu
Sent 2	English	Welcome to eCitizen
	Source	Karipu mu eCitizen
	System	Karipu to eCitizen
Sent 3	English	The boy is going to school
	Source	Omusoreri khacha esisomelo
	System	omusoleli khacha sikuli

Sent 4	English	Hello, my name is Velma.
	Source	Oriena, Lisina liange bali Velma
	System	Oriena my name esese velma
Sent 5	English	The cow is drinking milk
	Source	Ekhafu khenywa kamabele
	System	ekhafu esese khenywa kamabele

Table 6: Table showing sample translation results of the evaluation set

4.4 Discussions

Cross-lingual word embeddings were used to perform the unsupervised Neural Networks Machine translation. The first step in this exercise was initialisation, whose goal was to interrelate the English input and Bukusu output in an unsupervised manner by learning a mapping of the two languages. Unigram counts were extracted from the text data and then used to establish the word co-occurrence statistics in the dataset. This was also used to analyze the context in which a word is most often used by considering its neighbouring words, thus having the representation of the words in a vector space.

The vector representations were then used to train the translation models and test them. From the above results in Section 4.3 Model Performance Results, it is observed that the models performed poorly with little data, with most words being unique in the sentences. This meant the word embedding learning from English to Bukusu was not accurate enough to give the translation results. The models were retrained with additional data. It was observed that Encoder-Decoder and Bidirectional Recurrent Neural Network algorithms performed best among the four deep learning models used. Focusing on the two algorithms, they were subjected to different epoch values and batch sizes, then the accuracy and time taken were evaluated.

The training and validation data were split with an 80:20 ratio for each epoch. The loss, accuracy, validation loss, and validation accuracy metrics were recorded. It was noted that with increasing epochs, the loss value decreased and the accuracy increased. It was also noted that the validation accuracy increased and validation loss decreased, signaling that the model builds were learning and working fine without overfitting.

```

0.9318
Epoch 12/20
104/104 [=====] - 62s 598ms/step - loss: 0.4117 - accuracy: 0.9319 - val_loss: 0.4007 - val_accuracy:
0.9323
Epoch 13/20
104/104 [=====] - 66s 641ms/step - loss: 0.4022 - accuracy: 0.9320 - val_loss: 0.3939 - val_accuracy:
0.9322
Epoch 14/20
104/104 [=====] - 63s 605ms/step - loss: 0.3944 - accuracy: 0.9322 - val_loss: 0.3885 - val_accuracy:
0.9321
Epoch 15/20
104/104 [=====] - 61s 586ms/step - loss: 0.3861 - accuracy: 0.9324 - val_loss: 0.3906 - val_accuracy:
0.9322
Epoch 16/20
104/104 [=====] - 54s 518ms/step - loss: 0.3813 - accuracy: 0.9324 - val_loss: 0.3730 - val_accuracy:
0.9324
Epoch 17/20
104/104 [=====] - 50s 482ms/step - loss: 0.3799 - accuracy: 0.9325 - val_loss: 0.3747 - val_accuracy:
0.9324
Epoch 18/20
104/104 [=====] - 58s 556ms/step - loss: 0.3714 - accuracy: 0.9327 - val_loss: 0.3648 - val_accuracy:
0.9325
Epoch 19/20
104/104 [=====] - 55s 532ms/step - loss: 0.3642 - accuracy: 0.9329 - val_loss: 0.3578 - val_accuracy:
0.9332
Epoch 20/20
104/104 [=====] - 50s 479ms/step - loss: 0.3633 - accuracy: 0.9329 - val_loss: 0.3530 - val_accuracy:
0.9332

```

Figure 18: Figure showing the metrics of the Training Phase of the Encoder-Decoder model

```

Epoch 21/20
104/104 [=====] - 100s 963ms/step - loss: 0.2830 - accuracy: 0.9393 - val_loss: 0.2120 - val_accuracy:
0.9546
Epoch 13/20
104/104 [=====] - 102s 983ms/step - loss: 0.2633 - accuracy: 0.9418 - val_loss: 0.1866 - val_accuracy:
0.9577
Epoch 14/20
104/104 [=====] - 104s 1000ms/step - loss: 0.2456 - accuracy: 0.9441 - val_loss: 0.1712 - val_accuracy:
0.9605
Epoch 15/20
104/104 [=====] - 107s 1s/step - loss: 0.2309 - accuracy: 0.9463 - val_loss: 0.1531 - val_accuracy: 0.
9643
Epoch 16/20
104/104 [=====] - 103s 989ms/step - loss: 0.2193 - accuracy: 0.9481 - val_loss: 0.1457 - val_accuracy:
0.9660
Epoch 17/20
104/104 [=====] - 99s 952ms/step - loss: 0.2086 - accuracy: 0.9499 - val_loss: 0.1311 - val_accuracy:
0.9684
Epoch 18/20
104/104 [=====] - 100s 959ms/step - loss: 0.1979 - accuracy: 0.9516 - val_loss: 0.1237 - val_accuracy:
0.9707
Epoch 19/20
104/104 [=====] - 100s 964ms/step - loss: 0.1908 - accuracy: 0.9530 - val_loss: 0.1124 - val_accuracy:
0.9727
Epoch 20/20
104/104 [=====] - 100s 964ms/step - loss: 0.1796 - accuracy: 0.9548 - val_loss: 0.1053 - val_accuracy:
0.9742

```

Figure 19: Figure showing the metrics of the Training Phase of the Bidirectional RNN model

The Encoder-Decoder and Bidirectional RNN were trained using minibatch stochastic gradient descent with Adam optimizer, SoftMax activation and a learning rate of 0.005. The word embeddings were 120-dimensional and were trained for 15 minutes - which is significantly lower than other translation models due to the small dataset size. After building the models, the random outputs were visualised to show their new mapping on a vector space. Some words had a null next to them, signifying that the model could not translate such words.

The performance of the models was evaluated by measuring the BLEU score, WER, SER and TER. The encoder-Decoder model attained the highest BLEU score; meaning the model had the highest number of similar matches between automatic and reference translations. On the other hand, Simple RNN had the lowest score, symbolising a lower degree of similarity with the reference translation. The WER, SER and TER metrics were also measured and noted to be lower in Bidirectional RNN and Encoder-Decoder

compared to the other models. This indicated that these models had a higher translation accuracy. This measurement however provided no details on the nature of translation errors and was therefore a complementary evaluation metric. The SER was significantly lower in the Encoder-Decoder model suggesting the Bidirectional RNN translation has a weakness in processing long sentences due to the limited capacity of the fixed-length vector representation it uses. Similarly, its performance was noted to degrade in instances where the number of null or unknown translations increased.

Translation	Approach	Score
English → Malayalam	Statistical	20.8
English → Ekegusii	Rule Based	80% accuracy
English → Swahili	Statistical	0.20
English → Amharic	Neural Network (Encoder - Decoder)	0.33
English → Tigrinya	Neural Network (Transformer)	0.23
English → Nynorsk	Neural Network (Encoder – Decoder)	0.16
English → Ukrainian	Neural Network (Encoder – Decoder)	0.61
English → Belarusian	Neural Network (Encoder – Decoder)	0.23
English → Doha	Neural Network (Encoder – Decoder)	0.59
English → Beirut	Neural Network (Encoder – Decoder)	0.81
English → Rabat	Neural Network (Encoder – Decoder)	0.74
English → Tunis	Neural Network (Encoder – Decoder)	0.46
English → Bukusu	Neural Network (Encoder – Decoder)	0.55

Table 7: Table showing the BLEU score comparison of different translations.

Compared with other published translation works, as per Table 5, the model performs modestly. The hypothesis is that the dataset used to train the model is relatively small compared to the other models, which were trained with hundreds of thousands of sentences. This limited the neural learning of the morphological structure of the Bukusu language hence causing significant challenges in generating morphologically correct sentences and thus affecting the score.

5 CONCLUSION

This section details the remarks centered on the research study questions, methodology and outcomes. It commences with a discussion of the research findings and concludes with the limitations and recommendations based on the findings.

5.1 Discussion

The study's main goal was to establish a model for translating English text to Bukusu, particularly in digital services for e-governance. To ensure that this is addressed, four objectives were derived and generated research questions, whose outcomes are detailed below.

5.1.1 Investigating the machine translation techniques currently applied in translating low-resourced African languages.

This addressed the question, “What are the machine translation techniques being used for low-resourced African languages?” to understand the current techniques and their limitations. Statistical Machine Translation has been common in translating African languages, but neural networks have gained popularity since 2019. The typical approach used is dependency parsing and cross-lingual word embedding. Low-resourced language researchers from other parts of the world also use Part-of-speech tagging to translate text to their local languages.

5.1.2 Investigating the factors affecting the implementation of automatic machine translation of low-resourced African languages.

This addressed the question, “What are the factors affecting the implementation of machine translation for low-resourced African languages?” to understand the issues around low-resourced language translation. African languages are numerous, complex, and low-resourced. From the study, it is noted that datasets required for machine translation are difficult to discover, and existing research is hard to reproduce. The existing resources for African languages are challenging to discover as minimum numbers are published in renowned journals nor indexed by research tools. Previous researchers have not shared their data publicly meaning one cannot easily reproduce or benchmark previous work to new machine translation techniques.

5.1.3 Collecting and analyzing data for NLP in the automatic machine translation model.

This research objective aimed to build a parallel English-Bukusu dataset and perform NLP tasks on it. There was no publicly available dataset nor corpus, but electronic materials were available. Data was collected from the Bukusu bible, the most extensive collection of electronic texts in Bukusu, and the English – Bukusu dictionary. The Bible text had a sentence-to-sentence alignment, but for the dictionary

text, the Bukusu words did not have direct word translation but instead had the English explanation of the word. Due to limitations in computing resources, the bible texts collected were the first five verses of each chapter.

The final data set contained 5,146 parallel sentences of English-Bukusu, a small dataset for NMT compared to the well-resourced models, such as English - French and English - German, that are trained on billions of sentences. The data was randomly split into training, test, and validation sets of 75%, 20% and 5%, respectively.

NLP tasks performed on the data included Removing HTML tags, non-ASCII symbols and punctuation marks from the text, removing numbers that marked chapters in the hymn books and Bible, converting the text to lower case, removal of punctuation marks, tokenising the text to break it into individual linguistic units. An attempt was made to perform Named Entity Recognition using unsupervised learning; however, the results were incorrect. This was attributed to the fact that Bukusu has a different structure from English, which was used as the base for this exercise, and data on Swahili NER, whose language structure is like Bukusu, is not publicly available.

5.1.4 Developing and validating a model for automatic machine translation from English to Luhya

The model was successfully developed using Encoder – Decoder. It had a hidden layer size of 128 and embeddings had 256 units. The training phase ran for 50 epochs in batches of 100. The ADAM optimizer was used with a constant learning rate of 0.0005 to update the model weights. The model evaluation was done using the BLEU score as the main evaluation metric and WER, SER, and TER complemented the results. The model scored the highest BLEU score of 0.55, just 0.05 shy off the median range of 0.6 – 0.7 that has been achieved by other researchers. Compared to similar research on low-resourced languages, it scored modestly but outperformed the translation of English to Kiswahili (0.20) using Statistical Machine Translation.

5.2 Research Contribution

The significant contribution done through this research is the creation of an English-Bukusu translation model using neural networks. The model has a pipeline that preprocesses an English sentence to the simplest standard form, then encodes the input for mapping to the corresponding cross lingual Bukusu embedding through unsupervised learning then decodes the output as Bukusu text. This research contributes to the studies done for Unsupervised Neural networks and Machine translation for low-resourced languages.

5.3 Limitations in Research

The critical barrier observed during the research period was limited published work on the Bukusu language. This was observed right from the data collection, where the data was scarce, and the available data mainly was a religious text which influenced the model training and testing. Therefore, the model can be applied to limited e-Governance text, e.g., simple portals but not extensive text such as translating the Kenyan constitution. Human experts will be required to evaluate the results and correct translation errors.

Another barrier identified was challenges for future work such as the development of preprocessing and alignment tools for small-scale datasets and the need for a general text evaluation set. This is a limitation as there is no reference benchmark to assess the model's performance. Despite this, since the study was a prototype and not an end product, it can aid in further research towards translation to the Bukusu language.

5.4 Recommendation for Future Work

I recommend that more work be done on translating low-resourced languages. The key initiative should be the creation of a publicly available corpus that will serve as a catalyst for research in this area. This resource limitation can benefit from having audio resources by implementing speech recognition and speech-to-text since the Bukusu language is mainly spoken; and the lack of standardization in writing negatively impacts the creation of standard reference sets and evaluation. This study could also benefit from having reference models for Bukusu Named Entity Recognition and Parts of Speech tagging to improve translation accuracy.

Since the Bukusu language structure is like Swahili, the critical focus should first be on developing open-source NLP tools for the Swahili language. With this, researchers Bukusu and other low resources in other East African Bantu languages can transfer the Swahili models and annotations to their respective languages.

REFERENCES

- Abney, S., & Bird, S. (2010). The Human Language Project: Building a Universal Corpus of the World's Language. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 88-97). Uppsala, Sweden: Association for Computational Linguistics.
- African Translation Solutions by Translate 4 Africa.* (2021). Retrieved from Translate 4 Africa: <https://www.translate4africa.com/>
- Alla, S. (2022). *Advanced Recurrent Neural Networks: Bidirectional RNNs*. Retrieved from Paperspace Blog: <https://blog.paperspace.com/bidirectional-rnn-keras/>
- Ammar, W., Mulcaire, G., Ballesteros, M., Dyer, C., & Smith, N. (2016). Many languages, one parser. . *Transactions of the Association for Computational Linguistics 4*, (pp. 431–444.).
- Apirak Hoonlor, B. K. (2013, October). *An Evolution of Computer Science Research**. Retrieved from Rensselaer: <https://www.cs.rpi.edu/research/pdf/12-03.pdf>
- Artetxe, M., Labaka, G., & Agirre, E. (2017). Learning bilingual word embeddings with (almost) no bilingual data. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, (pp. 451-462).
- Bharati. (2010). *Machine Translation System in Indian Perspectives*. Academia.
- Bowker, L., & Ciro, J. B. (2019). *Machine Translation and Global Research: Towards Improved Machine Translation Literacy in the Scholarly Community*. Bingley: Emerald Publishing.
- Christodouloupoulos, C., & Steedman, M. (2015). A massively parallel corpus: the bible in 100 languages. *Language Resources and Evaluation*, 375–395.
- Cracking the Language Barrier for a Multilingual Africa.* (2021). Retrieved from Knowledge 4 All Foundation Ltd: <https://www.k4all.org/project/language-dataset-fellowship>
- Cracking the Language Barrier for a Multilingual Africa.* (2021). Retrieved from Knowledge 4 All: <https://www.k4all.org/project/language-dataset-fellowship/>
- Crnkovic, G. D. (2003). *THEORY OF SCIENCE*.
- Doochin, D. (2019, April 13). *What Are The Languages Spoken In Africa?* Retrieved from Babel Magazine: <https://www.babel.com/en/magazine/languages-of-africa>
- Doochin, D. (2019, August 31). *What Languages Are Spoken In Kenya?* Retrieved from Babel Magazine: <https://www.babel.com/en/magazine/what-language-is-spoken-in-kenya>
- Duong, L. T. (2017). *Natural Language Processing for Resource-Poor Languages*. Melbourne, Australia: The University of Melbourne.
- George, T. (2021, December 6). *Exploratory Research | Definition, Guide, & Examples*. Retrieved from Scribbr: <https://www.scribbr.com/methodology/exploratory-research/#:~:text=Exploratory%20research%20is%20a%20methodology,can%20be%20quantitative%20as%20well.>
- GovernmentOfKenya. (2014). *Kenya Vision 2030*. Retrieved from Kenya Vision 2030: <http://vision2030.go.ke/>

- Hamilton, T. (2022, September 3). *Software Testing Methodologies: Learn QA Models*. Retrieved from Guru99: <https://www.guru99.com/testing-methodology.html>
- Introduction to Recurrent Neural Network*. (2018, October 30). Retrieved from GeeksForGeeks: <https://www.geeksforgeeks.org/introduction-to-recurrent-neural-network/>
- Irfan, M. (2017). *Machine Translation*. Islamabad: ResearchGate.
- Jiang, C., Yu, H.-F., Hsieh, C.-J., & Chang, K.-W. (2018). Learning Word Embeddings for Low-Resource Languages by PU Learning. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)* (pp. 1024–1034). New Orleans, Louisiana: Association for Computational Linguistics.
- Kim, J.-K., Kim, Y.-B., Sarikaya, R., & Fosler-Lussier, E. (2017). Cross-Lingual Transfer Learning for POS Tagging without Cross-Lingual Resources. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 2832–2838). Copenhagen, Denmark: Association for Computational Linguistics.
- Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., . . . Ondřej Bojar, A. C. (2007). *Moses: Open Source Toolkit for Statistical Machine Translation*. Prague: Association for Computational Linguistic.
- Kothari, C. R. (2004). *Research Methodology Methods & Techniques*. New Delhi: New Age International Limited.
- Masakhane*. (2021). Retrieved from Masakhane: <https://www.masakhane.io/>
- McDonald, R., Joakim, N., Quirmbach-brundage, Y., Goldberg, Y., Das, D., Kuzman, G., . . . Lee, J. (2013). Universal dependency annotation for multilingual parsing. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, (pp. 92-97).
- Nguyen, T. Q., & Chiang, D. (2017). Transfer Learning across Low-Resource, Related Languages for Neural Machine Translation. *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)* (pp. 296-301). Taipei, Taiwan: Asian Federation of Natural Language Processing.
- Okpor, M. D. (2014). Machine Translation Approaches: Issues and Challenges. *International Journal of Computer Science Issues*, Vol. 11, Issue 5, 159-165.
- Ostling, R., & Tiedemann, J. (2018). *Neural machine translation for low-resource languages*. Cornell University.
- Pauw, G. D., Wagacha, P. W., & Schryver, G.-M. d. (2011). Towards English - Swahili Machine Translation. *Research Workshop of the Israel Science Foundation: Machine Translation and Morphologically-rich Languages*.
- Peng, L. (2013). *A Survey of Machine Translation Methods*. Universitas Ahmad Dahlan.
- Regulation By The Communications Authority of Kenya*. (2018, July 27). Retrieved from Communications Authority of Kenya: <https://ca.go.ke/>

- Sanders, R. (2020, April 9). *Digital inclusion, exclusion and participation*. Retrieved from Iriss: <https://www.iriss.org.uk/resources/esss-outlines/digital-inclusion-exclusion-and-participation>
- Sawe, B. E. (2017, August 25). *What Languages Are Spoken In Kenya?* Retrieved from Word Atlas: <https://www.worldatlas.com/articles/what-languages-are-spoken-in-kenya.html>
- Sciforce. (2019, October 11). *NLP for Low-Resource Settings*. Retrieved from Medium: <https://medium.com/sciforce/nlp-for-low-resource-settings-52e199779a79>
- Sghaier, M. A., & Zrigui, M. (2020). Rule-Based Machine Translation from Tunisian Dialect to Modern. *24th International Conference on Knowledge-Based and Intelligent Information & Engineering* (pp. 310-319). Elsevier.
- Shi, S., Wu, X., Su, R., & Huang, H. (2022). Low-Resource Neural Machine Translation: Methods and Trends. *Association for Computing Machinery*. China: Beijing Institute of Technology.
- Shreya Srivastava. (2019, October 31). *Machine Translation(Encoder-Decoder Model)!* Retrieved from Analytics Vidhya: <https://medium.com/analytics-vidhya/machine-translation-encoder-decoder-model-7e4867377161>
- Sirbu, A. (2015). *THE SIGNIFICANCE OF LANGUAGE AS A TOOL OF COMMUNICATION*. Constanta, Romania: “Mircea cel Batran” Naval Academy Press.
- Wikipedia. (2021). *Google Translate*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Google_Translate
- Zeman, D., Univerzita, K., & Resnik, P. (2008). Cross-language parser adaptation between related languages. *Workshop on NLP for Less Privileged Languages* (pp. 35–42). IJCNLP-08.
- Zoph, B., Yuret, D., May, J., & Knight, K. (2016). Transfer Learning for Low-Resource Neural Machine Translation. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 1568-1575). Austin, Texas: Association for Computational Linguistics.
- Zou, W., Richard, S., CER, D., & Manning, C. (2013). 1393–1398. *Conference on Empirical Methods in Natural Language Processing*, (pp. Bilingual word embeddings for phrase-based machine translation.).