Edge NLP for Efficient Machine Translation in Low Connectivity Areas

Tess Watt, Christos Chrysoulas, Dimitra Gkatzia
School of Computing, Engineering & the Built Environment
Edinburgh Napier University
Edinburgh, Scotland
{t.watt, c.chrysoulas, d.gkatzia}@napier.ac.uk

***Abstract*—Machine translation (MT) usually requires connectivity and access to the cloud which is often limited in many parts of the world, including hard to reach rural areas. Edge natural language processing (NLP) aims to solve this problem by processing language data closer to the source. To achieve this, 100 sentence pairs were stored and processed on a Raspberry Pi, and a recurrent neural network (RNN) using the long short-term memory (LSTM) architecture was used for machine translation. We are focusing on translating between English and Hausa, a low-resource language spoken in West Africa. It was found that the developed prototype produced "good and fluent translations" with a training accuracy of 91%. The model also achieved a BLEU score of 73.5, compared to the existing models that have scores of 22.2 and below.**

Keywords—edge computing, computation offloading, artificial intelligence, machine learning

I. Introduction

The ability to translate is fundamental for survival and improving people's quality of life, but currently AI applications such as machine translation, require an internet connection to function. Additionally, a report by the United Nations (UN) states that 37% of the world's population have never used the internet [1]. This lack of accessibility means that many people in remote areas, such as West Africa, are unable to use vital services such as Google Translate.

NLP coupled with edge computing can help overcome connectivity issues by processing language data within the device itself, even without internet access. This presents an exciting opportunity to make translation accessible to those with limited internet access, with a focus on the low-resource Hausa language, spoken in West Africa.

To address the need for transferring the large amount of data that is needed for MT tasks, we use edge computing which allows data to be offloaded to an edge device and remain there for the entirety of processing. To solve the problem of translation systems being inaccessible to people in low-connectivity areas, and the problem of big data putting a strain on cloud services, a Hausa-English translation system on the edge was developed.

Our main contributions are the following:
1) We present the architecture design for an edge NLP prototype based on device capacity and state-of-the-art machine translation methodologies.
2) We introduce a novel application of machine translation for low-resource situations.
3) We provide comprehensive experimental evaluation and comparative assessment of our approach against existing machine translation models.

The rest of the paper is structured as follows. Section II reviews related work. Section III describes our proposed system architecture and implementation, while Section IV provides a discussion of the results and comparative assessment, and finally Section V concludes our work.

II. Related Work

A. Cloud Computing

Cloud computing has been defined in [2] as a universal computing model for enabling on-demand access to a shared pool of resources. Typically, cloud computing is carried out via remote, centralised data centres. Some of the most well-known cloud services being Amazon Web Services (AWS), Microsoft Azure, and Google Cloud [3]. However, this traditional approach to cloud computing is becoming increasingly unable to handle the massive distribution of big data and the processing required to analyse it [4].

The challenges that centralised cloud computing is facing were explored in depth in [5] that within the current cloud computing infrastructure, resources are deemed as infinite. Despite the perceived unlimited computing resources, centralised cloud environments are experiencing serious problems with large data movements and latency, especially when it comes to the internet of things (IoT) [5]. The researchers in [6] claim that the IoT market is being driven by the rapid increase in the number of artificial intelligence (AI) assistants. AI assistants require sophisticated language models with a vast number of parameters and a high demand for memory [7]. Considering this, and the latency issues cloud has in IoT scenarios, the cloud may not be the best place to host AI assistants.

B. Edge Computing

Edge computing has been presented as a computing model that enables data to be stored and processed closer to the source from which it was gathered [8]. Edge devices (also known as 'low resource' devices) can range from hardware as small as Raspberry Pis and mobile phones to more standard hardware such as servers [5]. There appear to be no offline translation systems (deployed on edge devices) on the market that support Hausa. This gap in the market presents an opportunity for edge computing to be utilised.

According to [9] it is not always efficient for mobile users to retrieve localised data from the cloud. This is certainly the case with Google Translate, which to use offline, one must first connect to the internet to download the specific translation file to use offline later. Edge computing can overcome this inefficiency as it can provide services to users without relying on the cloud or the internet [10].

Cost is another clear benefit of using low-resource devices instead of expensive cloud servers. This is well supported by the work presented in [8] which states that one of the reasons edge computing has attracted so much interest over recent years is because of its low communication costs. These communication costs are low due to the ability of edge nodes to pre-process vast amounts of data before it is sent to the cloud. This is potentially beneficial to those who speak Hausa, as it is most widely spoken in West Africa, which has a high poverty rate.

It is important to note that the presented literature does not classify edge computing as a direct competitor to cloud, but as a model that can be utilized for use cases where cloud computing alone is not sufficient [5] - neither are mutually exclusive. To this end, the dominant cloud computing providers such as AWS and Azure are increasingly offering edge computing services for IoT [5].

C. *Computation Offloading Approaches*

With the integration of AI assistants and other IoTs into our everyday lives, the capacity required for storing the necessary data is challenging [11]. This is well supported in [5], which states that uploading such vast amounts of data to the cloud can be done, but should be avoided, as latency becomes a concern. Edge computing provides a solution to this, as data is processed within the device itself and therefore doesn't need to be moved to the cloud. This can be summarized in Ferrer's [5] claim that the primary aim of edge computing is to address cloud computing's latency issues in large IoT scenarios.

There are multiple benefits of implementing complex NLP models on edge devices over cloud including speed, reliability, security, privacy, and energy-saving [12]. Despite these benefits, it is argued that due to their large number of parameters and high demand for memory, sophisticated language models should not be hosted on edge devices in their standard form [7]. To this end, a range of methodologies have been developed to make these NLP models less memory intensive. The work presented in [13] began exploring this by introducing the concept of model compression, which later led to more specific techniques.

A more specific method of model compression is knowledge distillation (KD), which was first proposed in the work of Hinton et al. [14]. Knowledge distillation is a method of transferring knowledge from a large model called the teacher to a smaller model called the student [7]. This ability to transfer knowledge from a large model to a smaller one without losing validity makes KD ideal to be deployed on edge. Hinton et al. [14] showed that KD can be represented based on the following equation 1:

$$qi = \exp(zi/T)/\Sigma j \exp(zj/T) \quad (1)$$

Where a class probability (qi) is calculated by comparing the student logit (zi) with the teacher logit(s) (zj), and T being an optional 'temperature' value that is normally set to one. This is illustrated in Figure 1 which depicts a theoretical predicted translation of the Hausa word 'gobara' meaning 'fire'. This example of a neural network is highly confident that the translated word is 'fire', however it thinks there is a slight probability it could be 'flame'. This information would not have been extracted if hard one-hot encoding labels were used instead (where the probability of the word being 'fire' would be either 0 or 1) [15].

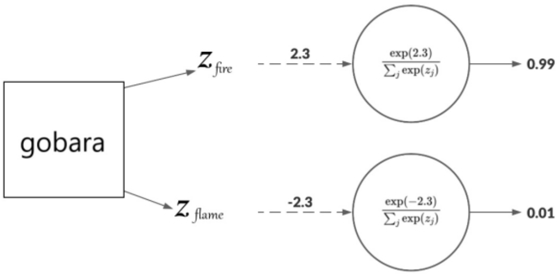

Fig. 1. Highly confident predictions of Hausa to English. Adapted from [15].

Setting the temperature to a higher value produces a 'softer' probability distribution over the class, giving more insight as to which classes the teacher found more equivalent to the predicted class [16]. This is illustrated in Figure 2 where the temperature value is set to five.

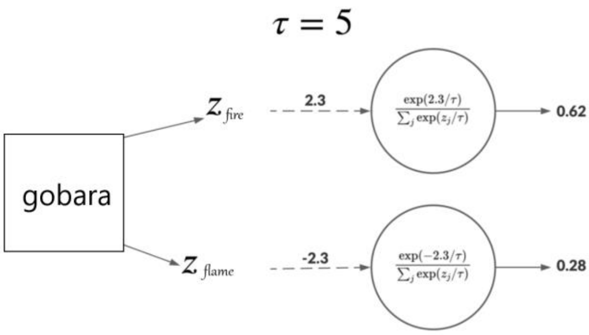

Fig. 2. Softened predictions of Hausa to English. Adapted from [15].

This is useful for small datasets, but for larger datasets (such as required for MT models) more information invariably becomes available by having more training examples [14]. Additionally, in [14], it is mentioned that increasing the temperature value is only useful to a certain extent. For example, in neural networks with 300 or more units in each of its two hidden layers, setting the temperatures above eight gave similar results. Ultimately, it was found that by using KD, the testing accuracy of their NLP model improved compared to their baseline model which simply used hard labels for training. The authors in [7] found that by using KD, the performance of two NLP models created for processing Modern Greek improved.

D. Neural-Based Machine Translation Approaches

Nwafor & Andy [17] explored how the neural-based approach has been used specifically to process Nigerian languages, including Hausa. Their paper is based on the work of Nguyen & Chiang [18], who created a model that focuses on translating rare words correctly. The authors in [18] claim that what makes neural machine translation (NMT) appealing is their end-to-end, singular-model training process and their performance compared to the statistical-based approach. The model they used was a combination of FixNorm [19] and 'lex', a simple lexical model developed using a feedforward neural network (FFNN) [18]. The results of this model were compared to existing literature. The first comparison drawn was to Arthur et al. [20], who created an MT model using discrete probabilistic lexicons. The second comparison drawn was to Moses, a phrase-based machine translation (PBMT) system [21]. Nguyen & Chiang [18] outperformed Arthur et al. on all tasks, and outperformed Moses on all tasks except for Urdu-English and Hausa-English.

Nguyen & Chiang [18] also found from these results that NMT tends not to perform as well as PBMT in low resource situations but concluded that NMT coupled with their model makes it a more viable option for low-resource translation than before. Considering all the above points, FixNorm + 'lex' may be a promising model for creating a well-performing Hausa-English translation system.

E. Data Privacy

A documented benefit of edge computing is data privacy. If data is not stored on the cloud, then it can be more difficult to reveal. This is supported by the work presented in [11] which states that as data is stored and processed locally, transmission to the cloud is unnecessary and therefore protects data privacy. Since the General Data Protection Regulation (GDPR) in the European Union was instated, data security and privacy are of the upmost importance [11]. Based on the aforementioned work, due to the GDPR the standard way of transporting data to a data centre will soon face a privacy barrier. Offloading such large volumes of data across various edge devices is a potential solution to this problem.

However, the work in [8] points out that edge isn't infallible. In terms of standard mobile edge computing (MEC) practices, user data is stored at a centralised edge node, which can be a cause for concern. Especially as mobile users are likely to possess sensitive data such as sexual orientation, political viewpoint, health status and more. This is important to consider when deciding which datasets should be used for training AI assistants, and where the data is collected from. Therefore, it could potentially be less of a risk to data privacy to use a neutral, open-source dataset such as news reports.

To conclude, the related work presents the latest developments and research challenges for future edge NLP models. It considers the impact of these findings for other researchers working in the field, and does so by discussing cloud computing, edge computing, computation offloading approaches, neural-based machine translation approaches, and data privacy.

The evidence tends towards hosting AI assistants on edge devices to avoid the latencies and inefficiencies associated with cloud computing. The success of cloud-based systems such as Google Translate cannot be denied, but the fact they are internet-dependent is a major drawback, especially as it inhibits those in low-connectivity areas from using the service. However, it is important to note that according to the literature, edge and cloud are not mutually exclusive.

Edge NLP has already been applied to the translation of Modern Greek [7], and neural-based machine translation approaches for processing Nigerian languages [17] have been reviewed. Our work builds on the existing literature by combining these approaches to develop an edge NLP system for the translation of Hausa.

III. System Architecture

A key consideration for the architecture design of an edge NLP prototype was the storage capacity and processor speed of the edge device. It was important to examine the capabilities of various devices that could be used to deploy the machine translation model. Table I shows the devices that were considered. As machine translation models are memory-intensive and require data to be processed quickly, a Raspberry Pi 3 Model B was chosen as the edge device.

TABLE I. STORAGE CAPACITIES AND PROCESSOR SPEEDS OF EDGE DEVICES

Device	Memory	Speed
Arduino UNO R3	2KB	16MHz
Raspberry Pi 2 Model B	1GB	900MHz
Raspberry Pi 3 Model B	1GB	1.2GHz

Figure 3 gives a high-level overview of the proposed translation system.

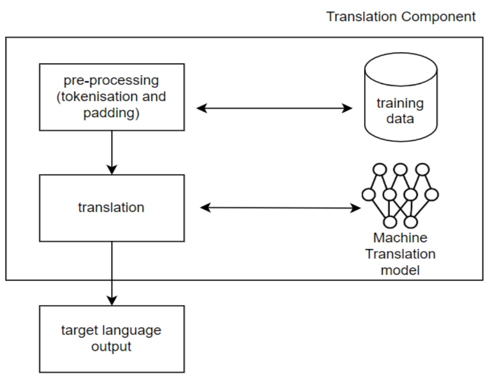

Fig. 3. High level architecture of the proposed Translation System.

The training data used for this system was 100 English-Hausa sentence pairs taken from open-source news data. The sentence pairs contained 1014 unique words in Hausa and 977 unique words in English [22].

The dataset of sentence pairs was labelled to allow for supervised learning to take place and was then analysed. The aim of this project was to create a machine translation model to determine what sentences in the source language would be correctly translated into the chosen target language (English or Hausa). This was achieved by changing the data representation, feeding it into a neural network, and evaluating the output.

The pre-processing carried out on the data before being fed into the machine translation model was tokenisation and padding. Tokenisation was used to split the text in the dataset into smaller pieces (individual words). Padding (specifically zero-padding) was used to maintain input dimensionality. Neural networks require inputs to have the same shape and size, but this is difficult when it comes to words and sentences as they are not all the same length. Zero-padding was used to solve this problem by adding zeros at the end of sequences to make them the same size. Word embedding was also used as it condenses more information into fewer dimensions than the bag-of-words technique, and words are represented using vectors. Word embedding is also specifically helpful for machine translation since it can determine similarities in word meaning [23].

The machine learning model used was a recurrent neural network (RNN) [24]. This was implemented using the Long-Short Term Memory (LSTM) architecture [25]. Language is sequential in nature as the order of words determines the meaning of sentences, hence the decision to use an RNN as they work well for processing sequential information [26]. Additionally, more cutting-edge models discussed in Section II are extremely complex and large, making an RNN a more suitable choice for use in practice on the edge. The model was adapted from an existing implementation [27], but several parameters were altered.

One of the parameters that was altered was the number of epochs. It was found that 700 epochs provided the best results (see Table II), as well as adding validation data. Out of all the different implementations tried, the one which provided the largest improvement in accuracy with the fewest number of parameters changed was implementation number 4 (see Table II), which used the Adam optimizer, softmax activation function, had a batch size of 64, and 700 epochs. After being trained once, this combination made 69% of translations correct.

TABLE II. RESULTS OF PARAMETER TUNING

Implementation Number	Validation Data	Optimizer	Activation Function	Batch Size	Epochs	Accuracy after being trained once (%)
1	No	Adam	softmax	64	200	69
2	No	Adam	softmax	64	500	69
3	No	Adam	softmax	64	700	70
4	Yes	Adam	softmax	64	700	69
5	Yes	Adam	softmax	128	700	70
6	Yes	Adam	sigmoid	128	700	70
7	Yes	SGD	sigmoid	128	700	69
8	Yes	Adam	sigmoid	256	700	70
9	Yes	Adam	sigmoid	512	700	69

IV. EVALUATION AND RESULTS

A considerable amount of parameter tuning took place to determine the optimal parameters for the model, although the difference between implementations ended up being only one or two percent. The best performing NMT model translated between English and Hausa with 69% accuracy after being trained once, 71% after being trained two times, and 91% after being trained ten times. The BLEU score after being trained ten times was 0.735 (73.5). A possible explanation for the model producing such good results on only 100 sentence pairs is the fact that Hausa contains many 'loanwords' from other languages, including English [28]. The model was also able to translate between English and Spanish as a comparative measure, with an accuracy of 98% after being trained once. Additionally, the model was successfully offloaded to the edge, without any excessively complex methods, such as knowledge distillation, being used.

Although the model achieved high quantitative training accuracies, the linguistic output often lacked quality. The predicted translation output at times contained missing and repeating words. To evaluate the linguistic quality of the translations produced, the outputs were shown to the members of Edinburgh Central Library's 'Found in Translation' book club, of whom several members are translators by profession. Members were asked to (anonymously) evaluate the quality of two Hausa-to-English translations using the Likert scale. Although there were mixed results, the most chosen Likert rating was 'Fair'. This further proves that despite the translations producing high evaluation metrics such as accuracy and BLEU score (high quality for a machine), the quality is not regarded as highly by humans.

To further evaluate the model, its architecture, parameters, and evaluation metrics, it can be compared to those in the related work. The NMT model developed by [20] was similar as it used the LSTM architecture. However, the experiments conducted for training took place on a GPU with 12GB of memory cache. This allowed their model to store and process a higher volume of training data, which contributes to higher accuracy predictions. Additionally, the BLEU score of the presented translation model was compared to those of the translation models discussed in the related work (see Table III).

TABLE III. COMPARISON OF TRANSLATION MODEL BLEU SCORES

Model	BLEU Score
Nguyen & Chiang	21.5
Arthur et al.	18.7
Koehn et al.	22.2
Proposed Model	73.5

As previously mentioned, the BLEU score for the presented translation model was 0.735. As BLEU scores are calculated between 0 and 1, it can be assumed this BLEU score has been expressed as a percentage (73.5%). According to [29] scores over 50 typically reflect good and fluent translations. However, it is difficult to evaluate whether the translation outputs are deemed as "good and fluent" by human evaluators, as the results gathered from the Found in Translation book club are mixed and subjective. For example, one member rated one of the translations 'Very Good' whilst two others rated the same translation as 'Poor'.

V. Conclusion

A Hausa-English translation system on the edge was proposed to solve two main problems: translation systems being inaccessible to people in low-connectivity areas, and big data putting a strain on cloud services. An appropriate architecture design for the prototype was created, with the relationship between cloud and edge computing being identified. Investigation into the most appropriate hardware to use as the edge device was conducted, with the conclusion that a Raspberry Pi 3 would be the most suitable candidate. The identified software components necessary for developing the translation system were discussed, and the system was evaluated against the results of existing models, producing a higher BLEU score after retraining.

As the presented model is currently trained on the edge, the next step would be to find a method of saving the pre-trained model and offloading it from the fog layer to the edge. This could potentially be achieved by saving the neural network weights to a joblib file - which would be offloaded to the edge device and loaded into the main program. As training will no longer take place on the edge, time will be saved, and the overall solution will be more efficient in terms of processing. Additionally, the future social implications of this research are promising. The Turing Trust have expressed interest in using the translation system as an Internet in a Box (IIAB) solution to be used in Africa.

References

[1] 'ITU: 2.9 billion people still offline | United Nations'. https://www.un.org/en/delegate/itu-29-billion-people-still-offline (accessed Jun. 20, 2023).

[2] Y. Guo, S. Wang, A. Zhou, J. Xu, J. Yuan, and C. Hsu, 'User allocation-aware edge cloud placement in mobile edge computing', *Softw Pract Exp*, vol. 50, no. 5, pp. 489–502, 2020, doi: 10.1002/spe.2685.

[3] J. Wang, J. Pan, F. Esposito, P. Calyam, Z. Yang, and P. Mohapatra, 'Edge Cloud Offloading Algorithms', *ACM Comput Surv*, vol. 52, no. 1, pp. 1–23, 2019, doi: 10.1145/3284387.

[4] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, 'Convergence of Edge Computing and Deep Learning: A Comprehensive Survey', *IEEE Communications surveys and tutorials*, vol. 22, no. 2, pp. 869–904, 2020, doi: 10.1109/COMST.2020.2970550.

[5] A. J. Ferrer, J. M. Marquès, and J. Jorba, 'Towards the Decentralised Cloud: Survey on Approaches and Challenges for Mobile, Ad hoc, and Edge Computing', *ACM Comput. Surv*, vol. 51, no. 111, 2019, doi: 10.1145/3243929.

[6] T. Tambe *et al.*, 'A 16-nm SoC for Noise-Robust Speech and NLP Edge AI Inference With Bayesian Sound Source Separation and Attention-Based DNNs', *IEEE J Solid-State Circuits*, pp. 1–13, 2022, doi: 10.1109/JSSC.2022.3179303.

[7] A. Goulas, N. Malamas, and A. L. Symeonidis, 'A Methodology for Enabling NLP Capabilities on Edge and Low-Resource Devices', in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2022, pp. 197–208. doi: 10.1007/978-3-031-08473-7_18.

[8] G. Liu, C. Wang, X. Ma, and Y. Yang, 'Keep Your Data Locally: Federated-Learning-Based Data Privacy Preservation in Edge Computing', *IEEE Netw*, vol. 35, no. 2, pp. 60–66, 2021, doi: 10.1109/MNET.011.2000215.

[9] H. A. Alameddine, S. Sharafeddine, S. Sebbah, S. Ayoubi, and C. Assi, 'Dynamic Task Offloading and Scheduling for Low-Latency IoT Services in Multi-Access Edge Computing', *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 3, pp. 668–682, 2019, doi: 10.1109/JSAC.2019.2894306.

[10] X. Ding and W. Zhang, 'Computing Unloading Strategy of Massive Internet of Things Devices Based on Game Theory in Mobile Edge Computing', *Math Probl Eng*, vol. 2021, pp. 1–12, 2021, doi: 10.1155/2021/2163965.

[11] S. Shen, Y. Han, X. Wang, and Y. Wang, 'Computation Offloading with Multiple Agents in Edge-Computing-Supported IoT', *ACM Trans Sens Netw*, vol. 16, no. 1, pp. 1–27, 2020, doi: 10.1145/3372025.

[12] V. S. Lalapura, J. Amudha, and H. S. Satheesh, 'Recurrent Neural Networks for Edge Intelligence: A Survey', *ACM Comput Surv*, vol. 54, no. 4, pp. 1–38, 2021, doi: 10.1145/3448974.

[13] C. Bucilǎ, R. Caruana, and A. Niculescu-Mizil, 'Model compression', *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, vol. 2006, pp. 535–541, 2006, doi: 10.1145/1150402.1150464.

[14] G. Hinton, O. Vinyals, and J. Dean, 'Distilling the Knowledge in a Neural Network', Mar. 2015, Accessed: Jul. 11, 2023. [Online]. Available: https://arxiv.org/abs/1503.02531v1

[15] 'Distilling Knowledge in Neural Networks With Weights & Biases | knowledge-distillation – Weights & Biases'. https://wandb.ai/authors/knowledge-distillation/reports/Distilling-Knowledge-in-Neural-Networks-With-Weights-Biases--VmlldzoyMjkxODk (accessed Jul. 12, 2023).

[16] 'Knowledge Distillation - Neural Network Distiller'. https://intellabs.github.io/distiller/knowledge_distillation.html (accessed Jul. 17, 2023).

[17] E. Nwafor and A. Andy, 'A Survey of Machine Translation Tasks on Nigerian Languages', 2022. [Online]. Available: https://www.cia.gov/the-world-factbook/countries/nigeria/

[18] T. Q. Nguyen and D. Chiang, 'Improving Lexical Choice in Neural Machine Translation', *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, vol. 1, pp. 334–343, Oct. 2017, doi: 10.18653/v1/n18-1031.

[19] Y. Zhou, Y. Sun, & Zhao, and Z. Huawei, 'FixNorm: Dissecting Weight Decay for Training Deep Neural Networks', Mar. 2021, Accessed: Jul. 11, 2023. [Online]. Available: https://arxiv.org/abs/2103.15345v1

[20] P. Arthur, G. Neubig, and S. Nakamura, 'Incorporating Discrete Translation Lexicons into Neural Machine Translation', *EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, pp. 1557–1567, Jun. 2016, doi: 10.18653/v1/d16-1162.

[21] P. Koehn *et al.*, 'Moses: Open Source Toolkit for Statistical Machine Translation'. pp. 177–180, 2007. Accessed: Jul. 11, 2023. [Online]. Available: https://aclanthology.org/P07-2045

[22] 'Translation Task - EMNLP sixth Conference on Machine Translation'. https://www.statmt.org/wmt21/translation-task.html (accessed Jul. 13, 2023).

[23] E. Rudkowsky, M. Haselmayer, M. Wastian, M. Jenny, Š. Emrich, and M. Sedlmair, 'More than Bags of Words: Sentiment Analysis with Word Embeddings', *https://doi.org/10.1080/19312458.2018.1455817*, vol. 12, no. 2–3, pp. 140–157, Apr. 2018, doi: 10.1080/19312458.2018.1455817.

[24] R. M. Schmidt, 'Recurrent Neural Networks (RNNs): A gentle Introduction and Overview', Nov. 2019, Accessed: Jul. 20, 2023. [Online]. Available: https://arxiv.org/abs/1912.05911v1

[25] S. Hochreiter and J. Schmidhuber, 'Long Short-Term Memory', *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/NECO.1997.9.8.1735.

[26] 'Text Classification — RNN's or CNN's? | by Shreya Ghelani | Towards Data Science'. https://towardsdatascience.com/text-classification-rnns-or-cnn-s-98c86a0dd361 (accessed Jul. 12, 2023).

[27] 'How to build an encoder decoder translation model using LSTM with Python and Keras | by Nechu BM | Towards Data Science'. https://towardsdatascience.com/how-to-build-an-encoder-decoder-translation-model-using-lstm-with-python-and-keras-a31e9d864b9b (accessed Jul. 17, 2023).

[28] P. Newman, 'Loanwords', *A History of the Hausa Language*, pp. 205–211, Mar. 2022, doi: 10.1017/9781009128070.006.

[29] A. Lavie, 'Evaluating the Output of Machine Translation Systems', 2011.