2nd Annual Meeting of the ELRA/ISCA SIG on Under-resourced Languages (SIGUL 2023)
18-20 August 2023, Dublin, Ireland

A Transformer-Based Orthographic Standardiser for Scottish Gaelic

Junfan Huang,[1] *Beatrice Alex,*[2] *Michael Bauer,*[3] *David Salvador-Jasin,*[4] *Yuchao Liang,*[5] *Robert Thomas,*[6] *William Lamb*[7]

[1,2,3,6,7]University of Edinburgh, UK, [4]The Alan Turing Institute, UK, [5]Independent Researcher, China

w.lamb@ed.ac.uk

Abstract

The transition from rule-based to neural-based architectures has made it more difficult for low-resource languages like Scottish Gaelic to participate in modern language technologies. The performance of deep-learning approaches correlates with the availability of training data, and low-resource languages have limited data reserves by definition. Historical and non-standard orthographic texts could be used to supplement training data, but manual conversion of these texts is expensive and time-consuming. This paper describes the development of a neural-based orthographic standardisation system for Scottish Gaelic and compares it to an earlier rule-based system. The best performance yielded a precision of 93.92, a recall of 92.20 and a word error rate of 11.01. This was obtained using a transformer-based mixed teacher model which was trained with augmented data.

Index Terms: Scottish Gaelic, text standardisation, text normalisation, transformer, neural network, Natural Language Processing (NLP), machine learning

1. Introduction

Text standardisation is the process of converting historical or orthographically irregular text to modern spelling conventions. There are two main motivations for conducting text standardisation with low-resource languages. The first is applying Natural Language Processing (NLP) tools developed for standard text in a language to historical or irregular text. The second is including historical or non-standard text as training data for tasks involving standard text. The latter motivation is key for low-resource languages such as Scottish Gaelic (henceforth 'Gaelic'), which by definition have a paucity of training data.

Gaelic is an endangered language spoken by roughly 57,000 individuals in Scotland. Unlike Irish, which was officially standardised in 1958, Gaelic has never had an official spelling system. The Gaelic Bible provided a *de facto* reference point until 1978, when the first organised attempt at standardisation occurred. This was led by the country's exam board, and culminated in the Gaelic Orthographic Conventions (GOC), which was adopted by schools and publishing houses and has been updated periodically since then [1]. In this paper, we report on an effort to develop a transformer-based system that automatically converts historical or irregular texts to GOC. Such a system, once sufficiently accurate, could help to maximise the training data available for Gaelic in a range of tasks including language modelling [2], part-of-speech tagging [3] and automatic speech recognition [4].

2. Related work

Text standardisation has been applied to historical text in languages such as English [5], French [6], German [7, 8], Irish [9], Portuguese [10] and Slovene [11] to name but a few. In early work, researchers tended to adopt rule-based and edit-distance-based methods [12, 13, 14, 15, 16, 17].

More recently, the field has moved to machine learning, often configuring the task as statistical machine translation (SMT) [9, 5]. Modern approaches are mostly neural-based [18, 19, 20]. Bollmann [21] conducted a comprehensive study on how these different techniques compare against gold standard data for eight languages and shows that SMT or neural algorithms outperform all non-learning-based methods. Scannell [9] incorporated Gaelic data within a statistic model used to standardise pre-modern Irish text, but the current publication is the first, to the best of our knowledge, that reports on automatic Gaelic text standardisation per se.

3. Data

Three datasets were used for the experiments described below: 1) pre-GOC and post-GOC aligned sentence pairs, 2) pre-GOC sentences from Corpas na Gàidhlig ('The Gaelic Corpus')[1] and 3) GOC-adherent Gaelic books, which were digitised by us and others. Together, they amount to 1,216,755 sentences or 18,080,158 words. These datasets are described further below.[2]

Dataset	Sents	Words	Characters	Sampling
Paired data	25k	310k	1.6M	100%
pre-GOC	918k	12.7M	67.8M	14.1%
GOC-books	272k	5.1M	27.2M	100%

Table 1: *Numbers of sentences, words and characters, and sampling percentage for each data type.*

Paired Data: Three groups of paired data were used for this research. The first is from the Calum MacLean project,[3] a repository of verbatim transcriptions of Gaelic folklore from the 1940s and 1950s. A Gaelic domain expert semi-automatically standardised these texts to conform to GOC, producing 25,417 paired pre-GOC and post-GOC sentences. The second group is

[1] https://dasg.ac.uk/about/cnag/en

[2] In this paper, we refer to all non-GOC texts as 'pre-GOC', although some of these texts post-date GOC's implementation.

[3] https://www.calum-maclean.celtscot.ed.ac.uk/calmac/home.htm

10.21437/SIGUL.2023-23

from the online Gaelic dictionary, Am Faclair Beag[4] and contains approximately 100k raw words (including headwords, idioms, and examples of usage). The third group, totalling 52,280 words, came from several out-of-print Gaelic fiction and non-fiction books that had been standardised and re-published by the team's Gaelic domain expert.

pre-GOC (CnGP): This data came from older, out-of-copyright texts that were digitised by the University of Glasgow as part of the Corpas na Gàidhlig project. The aggregate corpus totals 12,385,113 words. For the purposes of this study, we employed a strategic sampling approach, which allowed us to analyse 14.1% of the total corpus (15,000 sentences). This strategy was undertaken to achieve a balanced comparison with a real paired dataset consisting of 25,000 sentences.

Post-GOC: The Gaelic domain expert scanned, recognised and corrected a number of recently-published, GOC-adherent books of fiction and non-fiction. Other GOC-adherent texts came from authors directly, as well as from transcriptions of speech [4] linked to the ethnographic projects Guthan nan Eilean ('Island Voices')[5] and Tobar an Dualchais / Kist o Riches.[6]

4. Methods

In this section, we introduce our methodology, model architecture and experiments. The principal aim of this project was to build a transformer-based system that would outperform our earlier rule-based system [22]. The secondary aim was to offer our best performing models via a bespoke Gaelic text standardisation web app named 'An Gocair'.[7]

4.1. Data Processing

Data processing was executed using an automated pipeline using GitHub Actions workflows. This pipeline was designed to handle various file types, including .odt, .txt, and .doc, running our preprocessing script as the repository was updated. The Python package `langdetect` was employed to exclude English text from the training dataset, thereby focusing our model's training on Gaelic.[8] This workflow[9] ensured efficiency, reproducibility and a clear focus on our data processing tasks.

4.2. Data Augmentation

To amplify our limited initial data for the transformer model (see Section 4.3), we applied 5 techniques to create a synthetic parallel corpus of pre- and post-GOC text, enlarging our training data by a factor of 10.

Augmentation Techniques: Our approach comprised the injection of random noise, OCR noise, and pre-GOC noise to sentences from recent GOC-compliant literature. First, we introduced random noise – insertions, deletions, and character replacements – following the formula:

$$P(noise) = \frac{len(s) \times (ratio / \max - text - length)}{100}$$

where s represents a sentence, with the ratio defaulting at 0.6 and max_text length referring to the length of the longest sentence. This method was designed to disproportionately affect longer sentences and thereby create varied noise levels. The data was further enhanced by incorporating OCR noise, with typical OCR errors reintroduced to digitised texts post semi-automatic correction, facilitated by Python package `nlpaug` [23].[10] For pre-GOC noise, we applied replacement rules (e.g. à,è,ò → á,é,ó) and typographical errors to the texts[11], e.g. random *ll* to *l* swaps in non-initial contexts (e.g. *baile* → **baille*). In addition to these, we selected 15,000 pre-GOC sentences for rule-based data augmentation, which we termed 'Mixed Teacher', owing to its combined use of rule-based and human labels. We further employed a novel method, 'sentence random cropping' [24]. This was applied to 1% of the data, segmenting sentences to expose the model to words in varying positions and structures. To improve coverage of names, weekdays, months and numbers[11], we also generated an additional 40k sentences of artificial GOC-adherent text based on predefined rules. Finally, we included 420 lines of random special characters, floating-point numbers, and white-space to acclimatise the model to unseen tokens.

Following the processes of augmentation, duplicate removal, and exclusion of 1,067 non-compliant sentences, the resulting dataset consisted of 352,222 training sentences, collectively comprising 33 million tokens.

Method	Noise level	Tokens	Source
Random noise	23.4%	9.9 M	Post-GOC
OCR noise	23.4%	7.2 M	Post-GOC
Pre-GOC noise	5%	9.9 M	Post-GOC
Rule Based	-	1.8 M	Pre-GOC
Random Cropping	100%	0.3 M	Paired Data

Table 2: *Overview of data augmentation techniques, corresponding noise levels, number of tokens, and source data*

4.3. Model Architecture and Training

We utilised an encoder-decoder transformer architecture with 4 layers and 8 attention heads. With this, we trained a character-based transformer model using the Adam optimiser ($lr = 0.001, \beta_1 = 0.9, \beta_2 = 0.997$). The vocabulary was constructed using the post-GOC data along with some special characters. As proposed by Vaswani et al. [25], we also used label smoothing of 0.2 for regularisation. Beam search [26] was employed by default to generate the final output. The models were trained on a high-performance parallel compute cluster with four NVIDIA TITAN X GPUs. They were implemented in PyTorch using Fariseq, an open-source sequence modelling toolkit [27].

5. Evaluation

5.1. Error & Correction Rate Evaluation

We evaluated our methods on four evaluation scenarios: the real test set of 674 sentences, random noise, OCR noise and pre-GOC rule noise (see Tables 3 and 4).

[4] https://www.faclair.com
[5] https://guthan.wordpress.com
[6] https://www.tobarandualchais.co.uk
[7] https://angocair.garg.ed.ac.uk
[8] https://pypi.org/project/langdetect/
[9] https://bit.ly/git-repo-datapipe-line
[10] https://github.com/makcedward/nlpaug
[11] https://bit.ly/pre-goc-rule

5.2. Test Data

We deployed two distinct test sets to ensure robust evaluation of our models' performance across a variety of textual scenarios.

- The **real data pairs** tested generalisation capabilities, offering insights into how the models handled texts analogous to the training data.
- The **synthetic test sets** tested the models' resilience against various types of noise, a critical attribute for handling OCR-processed or user-generated texts.

5.3. Standard Metrics

In our study, we initially employed Precision, Recall and Word Error Rate (WER). While useful, these metrics do not fully capture the nuances of our task. Precision and Recall, in particular, can be misleading as they also account for correctly spelled words, which are in the majority, potentially leading to overly optimistic results.

To better reflect our task of accurately correcting errors, we introduced Correcting Recall and Correcting Precision. Correcting Recall ($CR = \frac{C_{corr}}{E_{total}}$) measures the proportion of true errors accurately corrected, while Correcting Precision ($CP = \frac{C_{corr}}{E_{ident}}$) gauges the proportion of identified errors accurately corrected. Where C_{corr} represents the number of errors correctly corrected by the model, E_{total} represents the total number of actual errors in the text, and E_{ident} represents the total number of errors identified by the model. These metrics focus on the model's error correction capability, providing a more granular view of its performance, as they specifically evaluate the model's ability to detect and correct errors, rather than just its overall accuracy. This is crucial in our task, as it allows us to better understand and improve the model's error correction capabilities, which is central to effective text standardisation.

We rejected Character Error Rate (CER) due to its limitations. CER treats each character as an isolated entity, misaligning with linguistic constructs and failing to appropriately capture the word-level transformations that are crucial in text normalisation tasks. Also, given its high values for this task (commonly over 95%), CER offers little meaningful differentiation in performance levels. WER, in contrast, accurately encapsulates word-level changes typical in text standardisation tasks, making it a more suitable choice for evaluating our models' performance.

6. Results

6.1. Model Comparisons

Table 3 provides comprehensive comparison of our various models on our 674 sentence test set, including the Rule-Based Model, Mixed Teacher Transformer (MTT), MTT with augmentation, Pure-heuristic Transformer (CnG), and JamSpell,[12] a Gaelic spelling correction model (using Am Faclair Beag's lexicon). The performance of the MTT-augmented model stands out. Moreover, this superiority is further accentuated when we implement a top-10 sampling (pass@10) strategy, leading to an enhancement in the overall results.

Interestingly, the Pure-heuristic Transformer (CnG) model, which utilises pre-GOC data and applies a rule-based model for transfer learning without using human labels, still performs competitively against other models. This indicates the inherent effectiveness of the Transformer architecture.

In comparison, the JamSpell model, which relies on a lexicon and edit-distance, has the lowest performance across the board, with the highest WER of 25.18%. The results clearly suggest that the Mixed Teacher Transformer model, especially when using data augmentation, is effective in this standardisation task across various types of noise and that it outperforms other models tested in this study.

6.2. Different scenarios

It is worth considering how our top-performing model (MTT-augumented) coped with various noise types. This is depicted in Table 4. We can see that the model performs particularly well on OCR noise (WER = 7.80). The performance on the 'Real test set' shows slightly lower metrics compared to the OCR noise, but it is important to note that this test set comprised a different data source. The Gaelic domain expert designed it to provide a range of steep, but naturalistic challenges to the system. Thus, it is impossible to directly compare this test set to the others. Future evaluations could consider obtaining test sets from a variety of sources instead of random sampling, which would ensure a more balanced evaluation.

7. Web Application and Command-Line Interface

We have released the Gaelic Standardiser as an online web application (see Figure 1).[13] It currently provides two transformer models with slightly different capabilities, and each model generates the top three outputs for any input sentence. The source code has been made publicly available on GitHub.[14] In addition to the web application code, this GitHub repository contains a command-line interface (CLI) written in Python which permits running the Gaelic Standardiser from the terminal.

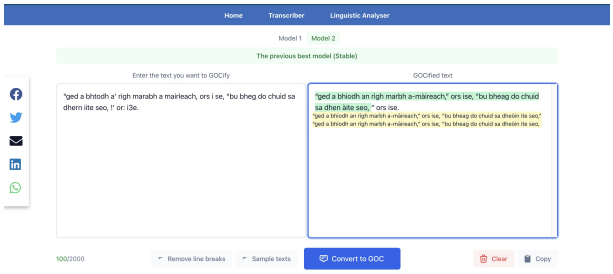

Figure 1: *Screenshot of the 'An Gocair' web app*

The web application front-end is based on a pre-existing paraphrasing tool.[15] Our front-end was built with Next.js, using the Create Next App tool[16] with a Type Script template. The back-end API is written in Python and employs the FastAPI web framework.[17]

Although this repository is still work in progress, it currently allows the user to deploy the web application locally and to run their own transformer model, in addition to the two models available in the online version. The user can run the FastAPI

[12] https://github.com/bakwc/JamSpell

[13] https://angocair.garg.ed.ac.uk/.

[14] https://github.com/Gaelic-Algorithmic-Research-Group/An-Gocair-Gaelic-Standardiser

[15] https://github.com/websymphony/paraphrasing-tool

[16] https://nextjs.org/docs/api-reference/create-next-app

[17] https://fastapi.tiangolo.com/

Table 3: *Results of Mixed Teacher Transformer on test set containing 674 sentences. We report correcting precision (CP), correcting recall (CR), precision (P) , recall (R) and word error rate (WER).*

Model	CP	CR	P	R	WER
Rule Based Model	65.96	56.35	94.01	91.26	12.18
Mixed Teacher Transformer (MMT)	57.04	57.19	91.23	91.27	13.61
MTT-augmented	**66.97**	**60.80**	**93.92**	**92.20**	**11.04**
MTT-augmented (pass@10)	79.62	72.28	96.39	94.59	7.10
Pure-heuristic Transformer (CnG)	64.02	54.03	93.68	90.73	12.98
Jamspell	29.84	18.18	89.20	81.19	25.18

Table 4: *The performance of Mixed Teacher Transformer (augmented data) model tested on different types of noise. We report correcting precision (CP), correcting recall (CR), precision (P), recall (R) and word error rate (WER).*

Noises	CP	CR	P	R	WER
Random Noise	66.58	64.66	95.64	95.27	11.16
Rule	87.32	76.93	96.65	93.32	9.43
OCR	94.06	87.39	98.69	97.06	7.80
Real testset	66.97	60.80	93.92	92.20	11.04

back-end application in the background using an ASGI server program like Uvicorn,[18] and access it through the locally deployed front-end. It should be noted that we have not made the pre-trained model files publicly available, so the user needs to provide their own if they would like to carry out a local deployment.

The Gaelic Standardiser back-end application and CLI make use of the PyTorch Fairseq toolkit.[19] The CLI can be employed to convert either a line of text, a single text file, or a folder containing text files, directly from the terminal.

8. Discussion & Future work

This paper presents initial experiments on the first neural network based models for Scottish Gaelic text standardisation. A character-based transformer model combined with data augmentation methods obtained the best performance. Ironically, given the motivation of our research, the biggest improvement in performance would come from having more paired sentences, as well as more GOC-adherent data overall. We hope to address this in the future and retrain our models using more input data. We also believe that enhancements could come from several other directions, as follows.

User Feedback Mechanism: One promising direction involves developing a web app with a user feedback mechanism. This would enable users to provide real-time feedback on the tool's outputs, which could be used to further improve its performance through Reinforcement Learning from Human Feedback (RLHF).

Generative Models: We intend to experiment with Generative Adversarial Networks [28] (GANs) or other generative techniques to create synthetic data. Such models have the potential to generate realistic and diverse Scottish Gaelic sentences, which could further enhance our data augmentation strategy and improve performance.

Advanced Data Augmentation Techniques: Although our current data augmentation strategy has proven effective, there is room for more advanced techniques. We have tried fine-tuning on a pre-trained model, but without seeing significant improvements. Future work could explore techniques like Elastic Weight Consolidation [29] (EWC), which allows the model to maintain its performance on the original tasks while learning new ones. Our plans for future work align with these strategies, seeking both to increase the quantity and quality of normalised Gaelic text available. It is hoped that these efforts – along with a range of other ongoing Gaelic speech and language processing projects – will contribute to the preservation and revitalisation of the language.

9. Limitations

Two key limitations in this work were the modest size of our training and test datasets. Further work should try to increase both sets. Due to the type of texts that are available in Scottish Gaelic, our input data also lacked diversity. For instance, a large proportion of it came from narrative texts – both fiction and traditional narrative. A further limitation is that we only used character-based models and not byte pair encoding (BPE) ones. While the latter tend to perform more accurately for machine translation tasks [30], they have been found to be less robust [31] and to perform worse on text normalisation [32].

10. Ethics Statement

This research gained institutional ethical approval on 5 Oct 2021. No substantial risks are associated with the work. Much of the training data came from in-copyright sources, however, and cannot be distributed.

11. Acknowledgements

This work was funded by the Scottish Government and Bòrd na Gàidhlig. We also gratefully acknowledge the Digital Archive of Scottish Gaelic (DASG: University of Glasgow) for providing training data, and the EPCC, the University of Edinburgh's high-performance computing centre, for their support.

[18] https://www.uvicorn.org/
[19] https://fairseq.readthedocs.io/

12. References

[1] Scottish Qualifications Authority, "Gaelic Orthographic Conventions," Report, 2009. [Online]. Available: https://www.sqa.org.uk/sqa/files_ccc/SQA-Gaelic_Orthographic_Conventions-En-e.pdf

[2] W. Lamb and M. Sinclair, "Developing word embedding models for Scottish Gaelic," in *Actes de la conférence conjointe JEP-TALN-RECITAL*, vol. 6, 2016, pp. 31–41.

[3] W. Lamb and S. Danso, "Developing an automatic part-of-speech tagger for Scottish Gaelic," in *Proceedings of the First Celtic Language Technology Workshop*, 2014, pp. 1–5.

[4] L. Evans, W. Lamb, M. Sinclair, and B. Alex, "Developing automatic speech recognition for Scottish Gaelic," in *Proceedings of the 4th Celtic Language Technology Workshop within LREC2022*, 2022, pp. 110–120.

[5] G. Schneider, E. Pettersson, and M. Percillier, "Comparing rule-based and smt-based spelling normalisation for English historical texts," 2017.

[6] R. Bawden, J. Poinhos, E. Kogkitsidou, P. Gambette, B. Sagot, and S. Gabay, "Automatic normalisation of Early Modern French," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 2022, pp. 3354–3366.

[7] N. Korchagina, "Building a gold standard for temporal entity extraction from Medieval German texts," 2016.

[8] E. Pettersson, "Spelling normalisation and linguistic analysis of historical text for information extraction," Ph.D. dissertation, Acta Universitatis Upsaliensis, 2016.

[9] K. Scannell, "Statistical models for text normalization and machine translation," in *Proceedings of the First Celtic Language Technology Workshop*, 2014, pp. 33–40.

[10] R. Marquilhas and I. Hendrickx, "Manuscripts and machines: the automatic replacement of spelling variants in a Portuguese historical corpus," *International Journal of Humanities and Arts Computing*, vol. 8, no. 1, pp. 65–80, 2014.

[11] N. Ljubešic, K. Zupan, D. Fišer, and T. Erjavec, "Normalising Slovene data: historical texts vs. user-generated content," in *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, vol. 16, 2016, pp. 146–155.

[12] H. Fix, "Automatische Normalisierung–Vorarbeit zur Lemmatisierung eines diplomatischen altisländischen Textes," *Teil*, vol. 3, pp. 92–100, 1980.

[13] A. Baron and P. Rayson, "Vard2: A tool for dealing with spelling variation in historical corpora," in *Postgraduate conference in corpus linguistics*, 2008.

[14] M. Kestemont, W. Daelemans, and G. De Pauw, "Weigh your words: Memory-based lemmatization for Middle Dutch," *Literary and Linguistic Computing*, vol. 25, no. 3, pp. 287–301, 2010.

[15] M. Bollmann, P. F., and S. Dipper, "Rule-based normalization of historical texts," *Proceedings of the International Workshop on Language Technologies for Digital Humanities and Cultural Heritage*, pp. 34–42, 2011.

[16] E. Pettersson, B. Megyesi, and J. Nivre, "Rule-based normalisation of historical text - a diachronic study," *11th Conference on Natural Language Processing, KONVENS 2012: Empirical Methods in Natural Language Processing - Proceedings of the Conference on Natural Language Processing 2012*, vol. 5, pp. 333–341, 2012.

[17] ——, "Normalisation of historical text using context-sensitive weighted levenshtein distance and compound splitting," in *Proceedings of the 19th Nordic conference of computational linguistics (Nodalida 2013)*, 2013, pp. 163–179.

[18] M. Bollmann, *Normalization of historical texts with neural network models*. Universitätsbibliothek Johann Christian Senckenberg, 2018.

[19] A. Robertson and S. Goldwater, "Evaluating historical text normalization systems: How well do they generalize?" in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 720–725. [Online]. Available: https://aclanthology.org/N18-2113

[20] M. Bollmann, N. Korchagina, and A. Søgaard, "Few-shot and zero-shot learning for historical text normalization," *arXiv*, pp. 104–114, 2019.

[21] M. Bollmann, "A large-scale comparison of historical text normalization systems," *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, vol. 1, pp. 3885–3898, 2019.

[22] R. Thomas, "'An Gocair': Gaelic normalisation at a click," 2021. [Online]. Available: https://blogs.ed.ac.uk/garg/2021/09/10/gaelic-text-normalisation-introducing-an-gocair/

[23] E. Ma, "NLP augmentation," https://github.com/makcedward/nlpaug, 2019.

[24] R. Takahashi, T. Matsubara, and K. Uehara, "Data augmentation using random image cropping and patching for deep CNNs," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 9, pp. 2917–2931, 2019.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[26] Y. Doval and C. Gómez-Rodríguez, "Comparing neural-and n-gram-based language models for word segmentation," *Journal of the Association for Information Science and Technology*, vol. 70, no. 2, pp. 187–197, 2019.

[27] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, "fairseq: A Fast, Extensible Toolkit for Sequence Modeling," in *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.

[28] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial nets ian," *Mining of Massive Datasets; Cambridge University Press: Cambridge, UK*, 2014.

[29] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[30] B. Heinzerling and M. Strube, "BPEmb: Tokenization-free pre-trained subword embeddings in 275 languages," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA), May 2018. [Online]. Available: https://aclanthology.org/L18-1473

[31] R. Gupta, L. Besacier, M. Dymetman, and M. Gallé, "Character-based NMT with transformer," *arXiv preprint arXiv:1911.04997*, 2019.

[32] I. Lourentzou, K. Manghnani, and C. Zhai, "Adapting sequence to sequence models for text normalization in social media," in *Proceedings of the international AAAI conference on web and social media*, vol. 13, 2019, pp. 335–345.