# A Generative Adversarial Approach to Training Data Synthesis for Natural Language Models

Natural Language Processing (NLP) has emerged as a transformative field that has revolutionized how machines can understand and interact with human language. However, progress in the field continues to be hindered by a scarcity of proper, subject-specific data. An increasingly popular solution to this problem involves the use of generative techniques to create synthetic datasets upon which natural language models can be trained. Despite greatly increasing the availability of training data, this comes at the cost of data quality; the lack of explainability of most generative methods makes the task of ensuring data validity near impossible. This research attempts bridge the gap by building upon existing work to apply Generative Adversarial Networks (GANs [1]) to Natural Language Generation (NLG) through rigorous data augmentation. Specifically, a modified GAN will be used to generate an initial sparse dataset with a diverse linguistic range, from which the data will be augmented to create a larger broad dataset suitable for training. By restricting the synthesis in this way, this will allow far greater control over the generated data than alternative methods (such as prompting a language model). The resulting model will be evaluated against several metrics (including BLEU [2], Zipf [3] coefficient, and others [4]) and then used to train an NL classifier. The study will conclude by outlining a set of guidelines and reasoning for effective NL data synthesis so that this approach may be reproduced for other domains.

## INTRODUCTION

The current and future influence of NLP technology is self-evident, with successful applications across several domains, whether as highly versatile virtual assistants (Siri [5], Google Assistant [6]) or as efficient translators (Google Translate [7]). The most recent advance in the field was the release of ChatGPT-4[8] which has brought NLG firmly into the public eye. An important factor in the success of all of these models is the vast amount of computational and human resources that their creators (Apple, Google, OpenAI, etc.) have at their disposal, which allows them to overcome the problems of data scarcity. This is of course, not possible for the overwhelming majority of projects and so work must be done to fine more efficient ways to train NLP models.

The inspiration for this research avenue came from the VR Study ("Creating Immersive Training Experiences in Virtual Reality" [9]) conducted as a vertically integrated project at the University of Bath. The goal of this study is to develop a VR simulator to train individuals to be effective bystanders when witnessing sexual harassment and to safely prevent and put a stop to such scenarios. The user does this by speaking to the perpetrator, which is then followed by the system classifying their utterance into one of 5 actions that the user can perform to diffuse the situation:

- Delay: respond afterwards, check on the victim – "[VICTIM NAME], are you okay?"
- Delegate: involve other people – "[FRIEND NAME], you have to say something, that's not right."
- Direct: use reasoning, refute comments/actions – "Come on, now, that's wrong."
- Distract: divert attention to objects or people – "Hey, let's talk about something else?"
- Document: record the dialogue (this an action and is not done by speaking, ignored by model).

While the implementation of such a model is simple in theory (the system uses a bidirectional LSTM [10]), manually obtaining a large/broad enough dataset to effectively train the model is impractical (given project resources) and there is no existing dataset given the specialised nature of the problem. In addition, the context of each scenario changes from scene to scene, meaning that scene-specific data would have to be manually collected. To bypass this, several attempts have been made to generate a sufficient dataset using a variety of LLMs (but primarily GPT-4[8]). Despite generating and training the LSTM on a total of 8000 synthetic utterances, the model continues to underperform; the LLMs generate sentences in a very specific style that is not representative of realistic human speech, even though efforts have been made to engineer the prompts to do otherwise.

Even worse, the LLMs generally misunderstood the meanings of the different actions (for example, much of the data for the "Distract" action was focussed on the user distracting themself and letting the situation unfold, rather than distracting the attacker). Of course, this is highly problematic as training the model on

such faulty data could result in the system incorrectly teaching the user and could have potentially disastrous consequences. Since one cannot manually check the validity of each data point, the use of these particular language models is hard to justify.

The goal of this research is to create a standardised approach to NLG for the purposes of synthesizing valid training data, using GANs coupled with intensive data augmentation. GANs are a well-known framework for image generation, however they have seen less success in NLG than techniques such as Variational Autoencoders [11] and autoregressive methods (Transformers [12]), due to the reduced structure and increased data representation complexity that comes with natural language. Due to their intrinsically continuous nature, GANs also struggle with processing the discreteness of language and text. Nevertheless, with proper performance measurement and data augmentation techniques, GANs may yet reach their full potential in the field of NLG.

The choice to use GANs for this research instead of alternative techniques (such as VAEs) was done for a number of reasons. First, the adversarial approach to learning in GANs means that they learn to mimic realistic data rather than an underlying distribution like VAEs, ensuring that the generated data is similar to the real labelled data. This also results in a more diverse linear interpolation of the latent space resulting in a sample that is broad yet sparse, from which we can use data augmentation to generate families of similar sentences. Finally, since the aim is to create an approach that is less limited by computational resources, GANs were chosen as they are generally easier to train (computationally speaking) than alternative methods.

## RELATED WORK

(Goodfellow et al., 2014) [1] needs no introduction as it was the first paper to propose the GAN framework. Unlike the majority of ML papers that propose a new technique and reinforce the claim with heaps of empirical evidence, this paper delves into rigorous mathematical detail to prove that the framework is sound and convergent. Whilst presenting less results than other papers, the quality of the results presented was a great improvement for the time and would set the standard for many years to come. Particularly, Figure 3 of the paper shows the linear interpolation of the z-space of an MNIST [13] digit generator. The presentation of this behaviour would become a staple of generative ML papers, and for good reason as this demonstrates that the generator displays some understanding of what constitutes each digit. Although it is much harder for GANs to learn the complicated relationships that constitute human speech, this research will attempt to perform semi-supervised learning with a GAN variant to create a sparse synthetic dataset, from which we can apply data augmentation to create a broader training dataset.

Building on this shortly after, (Mirza and Osindero, 2014) [14], implemented the idea of a conditional GAN (cGAN), which incorporates additional conditioning information to guide the generation, enforcing the synthesis of data samples that meet specific criteria. This information could be attributes, properties or in this case, classes. The experimental results based on the Gaussian Parzen window were relatively underwhelming, as the cGAN was outperformed by a number of techniques, some even non-adversarial. Nevertheless, the authors themselves note this experiment as a “proof of concept” and cGANs have since been shown to be highly effective in image generation and transformation [15] (although admittedly still not reaching the same success in NLG). While the other GAN variants that are discussed later will likely prove useful, this conditioning is essential and will be used for each model that is tested, given that our goal is to produce a labelled training dataset rather than an unlabelled list of sentences (most of the other GAN papers include a conditioned version as well as an unconditioned one). With a cGAN it is also possible to generate a certain amount of data for each of the 4 actions in the classifier, allowing greater control of the structure of the synthetic dataset.

However, first we must address the poor performance of GANs in a natural language context. A number of attempts have been made to modify the default GAN framework for such contexts, most of which propose more complex and meaningful ways of evaluating the generator’s performance.

The paper (Yu et al., 2017) [16] builds upon the GAN framework by introducing SeqGANs, the first attempt to apply the adversarial framework to sequential generation. Rather than using a binary discriminator (classifying data as 0 for fake and 1 for real), SeqGAN discriminators use reinforcement learning to evaluate the data: the feedback from the discriminator is in the form of several rewards and so the model is trained by maximising the expected cumulative reward instead of minimising a loss function. This provides

the generator with far more information about the abstract patterns of natural language, resulting in a higher quality generator output. The paper demonstrates this as the SeqGAN outperforms all tested alternative techniques.

(Fedus et al., 2018) [17] also uses RL to train the generator, however this paper specifically addresses the problem of exposure bias in sequential models: during training, the model is exposed to ground-truth tokens at every timestep whereas during inference the model is forced to rely only on previous tokens. To fix this discrepancy, the paper applies masking, which involves intentionally omitting input/output tokens. Not only does this simulate how the model would actually perform under inference, but it also encourages the generator to produce sentences that are realistic and coherent even when parts are obscured or missing, resulting in higher quality data. The experimental results support this as the MaskGAN dominates almost all categories and closely matching the SeqGAN. A particularly notable result was that masking tokens randomly produced lower quality results than masking tokens continuously. When discussing this, the authors hypothesize that this gives the generator more flexibility to appropriately fill in the gaps, resulting in larger performance increase.

(Arjovsky et al., 2017) [18] propose an alternative solution, Wasserstein GANs (WGANs). Unlike the previous two approaches that use RL-based evaluation, WGAN discriminators are based on Wasserstein distance rather than a traditional loss function. Wasserstein distance is a concept from transportation theory that measures the difference between distributions (specifically how much “work” must be done to transform one into another). This is a much more meaningful measure of how closely the generator can mimic the real data, allowing the framework to produce more accurate synthetic data. This training procedure is also much more stable than for regular GANs as it suffers less from mode collapse and vanishing gradients.

In order to ensure the stability of and convergence of the training, (Arjovsky et al., 2017) [18] attempted to enforce the Lipschitz constraint on the discriminator output (so that its output does not change drastically given small changes in the input). They originally did this using gradient clipping: limiting the magnitude of the discriminator’s gradients, so that the Wasserstein distance between the estimated and real distributions are well defined. However, as is typical with introducing such constraints, this approach exacerbates other problems such as that of vanishing and exploding gradients. (Gulrajani et al., 2017) [19] propose an alternative, softer method for maintaining the Lipschitz constraint called gradient penalty (GP). This method is conceptually similar to L2 regularization, except that the goal is to encourage a smoother gradient rather than smaller weights. As the name suggests, this method penalizes large gradients by adding a squared gradient term to the loss function, which will passively encourage this outcome. This approach resulted in a significant improvement in performance over the standard WGAN.

LeakGANs were proposed in (Guo et al., 2017) [20] as a technique for generating coherent and relevant text of increased length. This technique works by introducing a dynamic lexicon, which supplies the generator with additional context and information such as topic keywords to ensure that the generated text remains consistent throughout. Although effective, the data we hope to generate for our LSTM classifier will have a short sequence length and so this technique will not be used, however it should absolutely be considered when adapting this approach for other domains.

(Semeniuta et al., 2019) [4] attempts to adapt the GAN framework for NLG by questioning the ways that the framework’s performance is assessed. They argue that using purely N-gram-based metrics (such as BLEU) to evaluate network performance leads to the generator misrepresenting the real data. The study proposed alternate metrics such as Frechet InferSent Distance and even using an LM to score the output. These metrics were used to evaluate the performance of several GAN variants such as the SeqGAN [16] and LeakGAN [20], noting an increased performance with the new metrics. Despite this, the results showed that all GAN variants struggled to compete with even a simple LM. Unlike previous papers, this experiment gives some evidence conclude that while GANs can be adapted to better handle natural language, this may only go so far. It will be important to consider this limitation going forward.

Clearly, there are a number of ways that the GAN framework can be modified to generate the initial sparse dataset. We may now begin to discuss the many ways that this data could be augmented to achieve the desired training dataset.

To bridge the gap between simulated and real data, (Marzoev et al., 2020) [21] proposes the idea of a projection function to transfer knowledge from real data to synthetic paraphrase data (sentences that share

the same meaning). The technique works by training a model to create synthetic utterance data (in this case a GAN), training the model to interpret the synthetic data (the LSTM), whilst separately learning sentence representations (word embeddings) with an LM. To evaluate a human input, we use the learned sentence representations to project onto the set of all possible sentences achievable by the generator and then interpret that synthetic sentence with the NLP model. This way, we can limit our generator to produce a specific set of target sentences to which we can map a much larger and more diverse set of paraphrase data that covers most human speech patterns whilst staying within the domain of the problem. The paper found that the synthesized data was comparable to real supervised data. It is likely that the explicit encoding of some of the patterns present in natural language provide the model with a significant head start over models that would have to learn this behaviour without guidance.

Perhaps the most common method of data augmentation for natural language was presented by (Wei and Zou, 2019) [22] where they discuss synonym replacement (among other techniques). As the name suggests, this technique involves generating paraphrase data by replacing random words in the sentence with their synonyms. This is an incredibly simple yet effective technique for creating new sentences without changing their context (label). The paper combines synonym replacement with other techniques (such as random deletion/replacement) to form EDA. While EDA did show an increase in performance, the authors note that this increase is marginal for large datasets. The authors also remark that they do not expect for EDA to be used in practice as they expect others to propose new and improved methods for augmenting NL in the near future, however it seems that in retrospect little progress has been made since.

(Li et al., 2023) [23] emphasises the problem of subjectivity in natural language classification (where the classes are not only ambiguous but varying from person to person). This was demonstrated by the faulty data generated by the unspecialized LLMs (the “distract” example from the introduction was a textbook case of this). This study presents evidence that augmenting real data is more reliable than generating purely synthetic data from pretrained language models like GPT-4[6]. While there may be some truth to this, it may not always be possible if there is insufficient data even for that step.

In contrast to generating paraphrase data, (Kumar et al., 2022) [24] focusses on text style transfer (changing a sentence without changing the content). The paper greatly emphasises a distinction between spoken language and written language (something that unspecialized LLMs confuse frequently), even stating that the thoughtless removal of inaccuracies such as filler words or repetition in a spoken data will result in data that is much closer to a written style. This is suboptimal as for the LSTM, we only want to classify spoken text and so the training data would not be representative of the real inputs. Much of the paper is dedicated to identifying the style of a sentence (written or spoken) using a genre classifier and then using text style transfer for converting between spoken and written language styles which was especially successful. This study intends to generate purely spoken data and so the techniques described will likely not be used unless generating spoken data is more difficult than predicted.

(Kim et al., 2022) [25] proposes the SDI (Synthetic Data Identification) technique to classify real and synthetic sentences in a given dataset. It does this by identifying linguistic properties that are unnatural to be spoken by a human (such as repetition or logical flaws). From this, the paper proposes the RISE method which assigns a lower importance to unnatural sentences, allowing the training of more reliable sentence encoders. The paper shows promise as it managed to outperform models trained on unspecialised, unreliable sentences. While an interesting approach, this somewhat contradicts with the philosophy of (Kumar et al., 2022) [24] who argue that such inconsistencies will be made when the user’s speech is passed as input to the NLP system. To find a balance between these two sides, we must ask to what extent we want to train our model to accept and handle user failure, and at this point in the project it is not yet clear where the line should be drawn.

Turn over for REQUIREMENTS.

## REQUIREMENTS

| 1 | **HIGH PRIORITY** | **SUCCESS CRITERION** |
|---|---|---|
| 1.1 | Create a modified GAN that outputs VR study data. | Be able to synthesize training categorical data specific to VR study. |
| 1.2 | Data should be subject specific and realistic. | Data must perform well under SDI [25] method (less than 5% flagged as unrealistic). |
| 1.3 | Data should be used to train NLP classifier. | Successfully train the classifier on the synthetic dataset. Classifier should achieve at least 90% accuracy on real validation data. |
| 1.4 | Compare classifier trained on real, unspecific synthetic, and GAN synthetic data. | Be able to determine the best performing dataset/approach: record relevant metrics [2,3,4] and determine best model(s). |
| **2** | **MEDIUM PRIORITY** | **SUCCESS CRITERION** |
| 2.1 | Be able to produce realistic data by linearly interpolating z-space. | Sampled datapoints should be coherent, contextually relevant and gradually transforming from one to another (evaluate manually). |
| 2.2 | Focus on computational speed and memory usage. | Be able to determine most efficient approach (record synthesis time and memory usage). Aim to outperform LLMs such as ChatGPT [8]. |
| **3** | **LOW PRIORITY** | **SUCCESS CRITERION** |
| 3.1 | Be able to produce colloquial semantic data. | Data is indistinguishable from real colloquial language (evaluate using SDI [25], less than 5% flagged as unrealistic). |
| **4** | **PROBLEM CONSTRAINTS (NOT COVERED IN RESEARCH)** | |
| 4.1 | Be able to produce data of different speech patterns. | |
| 4.2 | Be able to produce colloquial semantic data. | |

Turn over for SOLUTION SKETCH.

## SOLUTION SKETCH

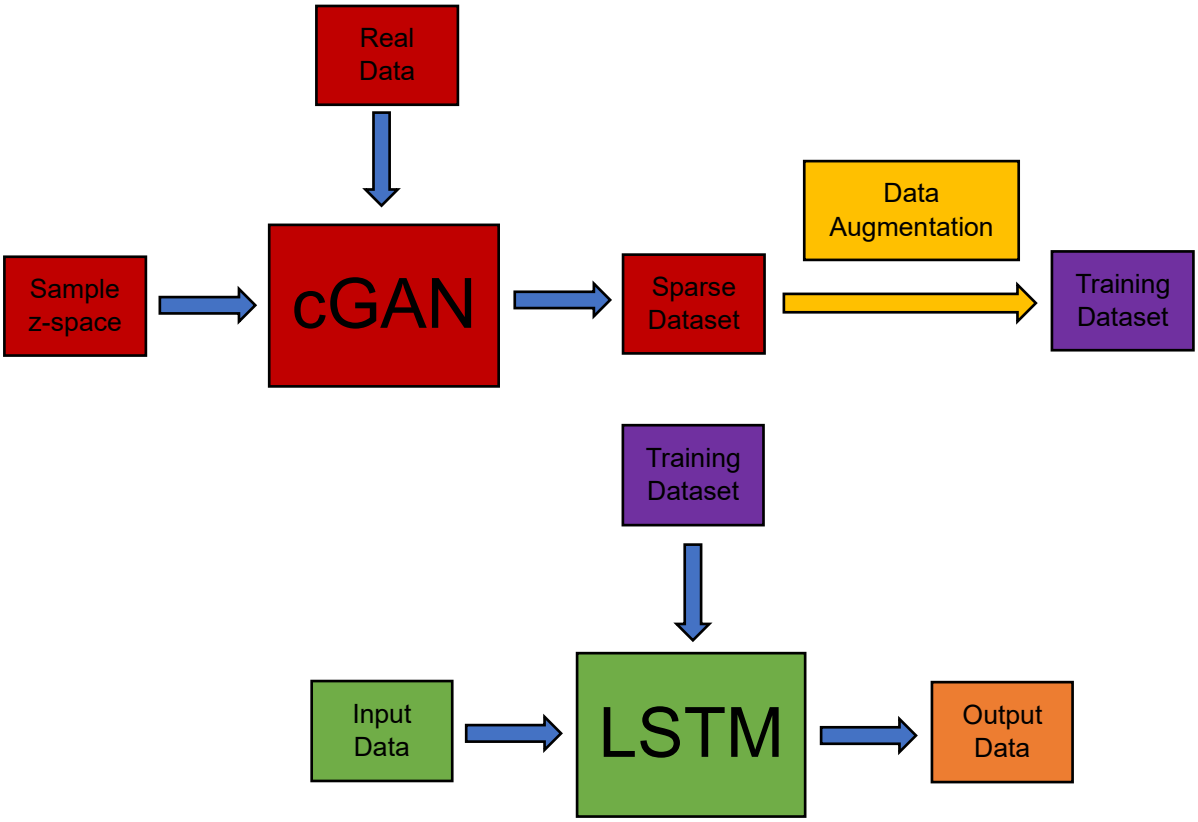


***Figure 1:*** *Proposed architecture of main idea: 1. A conditional GAN will be trained to synthesize supervised utterance data, 2. Sample generator z-space to produce a sparse, diverse dataset, 3. Use data augmentation techniques to synthesize parallel data and create final dataset, 4. Train NLP model on synthetic training dataset.*

Initially, the research will begin by recreating the GAN variants, including vanilla GANs, cGANs, SeqGANs, MaskGANS, WGANs and WGANs-GP (excluding LeakGANs). First, the variants will be evaluated manually, noting the behaviour, discrepancies and overall performance of each. Once this general understanding has been obtained, each variant will be evaluated based on a number of metrics, including BLEU, Zipf coefficient, and as suggested by (Semeniuta et al., 2019) [4] Frechet InferSent Distance and LM score. This is not only to further assess each variant's performance but also to gauge how well each of these metrics actually measure the quality of the synthesized data. Once this has been carried out, the focus will shift to evaluating the data augmentation techniques, specifically EDA, RISE and projection, using the same metrics as before and noting the changes. Eventually, the NLP model will be trained on each combination of GAN variant and augmentation technique, allowing for comparison and selection. At last, the validation accuracy for each NLP model will be collected to determine if such methods for training data synthesis are better than using unspecialised LMs.

## CONCLUSION

This study proposes the research into using adversarial techniques coupled with data augmentation to generate realistic and context specific data to be used in the training of NLP models. A standardised approach to synthesizing NL training data is long overdue as there are countless other customised, domain-specific problems that could benefit by adapting this technology. Similarly, this approach could be transferred over to other languages to help equalise the gap in training data across different languages (many languages suffer from a lack of training data [26]). For the purpose of specificity, this research will only focus on sentiment classification for the English language, however this proposal invites others to repurpose and build upon the approach for their problem domain and language.

## REFERENCES


1. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y., 2014. *Generative Adversarial Nets* [online] arXiv.org. Available from: https://arxiv.org/abs/1406.2661 [Accessed 22 Apr. 2024].
2. Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J., 2002. *BLEU: A Method for Automatic Evaluation of Machine Translation.* [online] ACM Digital Library. Available from: https://aclanthology.org/P02-1040.pdf [Accessed 22 Apr. 2024].
3. Zipf, G.K., 2012. *Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology*. Mansfield Centre, Connecticut: Martino.
4. Semeniuta, S., Severyn, A. and Gelly, S., 2019. *On Accurate Evaluation of GANs for Language Generation*. [online] arXiv.org. Available from: https://arxiv.org/abs/1806.04936 [Accessed 23 Apr. 2024].
5. Apple, 2011. *Siri*. [online] Siri - Apple. Available from: https://www.apple.com/siri/ [Accessed 22 Apr. 2024].
6. Google, 2016. *Google Assistant*. [online] Google assistant, your own personal google. Available from: https://assistant.google.com/ [Accessed 22 Apr. 2024].
7. Google, 2006. *Understand your world and communicate across languages*. [online] Google Translate. Available from: https://translate.google.com/about/ [Accessed 22 Apr. 2024].
8. OpenAI, 2023. [online] GPT-4. Available from: https://openai.com/gpt-4 [Accessed 22 Apr. 2024].
9. University of Bath, 2024. *Vertically Integrated Projects - Current VIPs*. [online] Vertically Integrated Projects - Current VIPs. Available from: https://www.bath.ac.uk/guides/vertically-integrated-projects-current-vips/#creating-immersive-training-experiences-in-virtual-reality [Accessed 22 Apr. 2024].
10. Hochreiter, S., and Schmidhuber, J., 1997. *Long Short-Term Memory*. [online] ResearchGate. Available from: https://www.researchgate.net/publication/13853244_Long_Short-term_Memory [Accessed 22 Apr. 2024].
11. Kingma, D.P., and Welling, M., 2022. *Auto-encoding variational Bayes*. [online] arXiv.org. Available from: https://arxiv.org/abs/1312.6114 [Accessed 24 Apr. 2024].
12. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L. and Polosukhin, I., 2023. *Attention is all you need*. [online] arXiv.org. Available from: https://arxiv.org/abs/1706.03762 [Accessed 24 Apr. 2024].
13. LeCun, Y., Cortes, C. and Burges, C.J.C., 1999. *The MNIST Database*. [online] MNIST Handwritten Digit Database, Yann LeCun, Corinna Cortes and Chris Burges. Available from: http://yann.lecun.com/exdb/mnist/ [Accessed 27 Apr. 2024].
14. Mirza, M., and Osindero, S., 2014. *Conditional Generative Adversarial Nets*. [online] arXiv.org. Available from: https://arxiv.org/abs/1411.1784 [Accessed 28 Apr. 2024].
15. Isola, P., Zhu, J.-Y., Zhou, T. and Efros, A.A., 2018. *Image-to-Image Translation with Conditional Adversarial Networks*. [online] arXiv.org. Available from: https://arxiv.org/abs/1611.07004 [Accessed 29 Apr. 2024].
16. Yu, L., Zhang, W., Wang, J. and Yu, Y., 2017. SeqGAN*: Sequence Generative Adversarial Nets with Policy Gradient*. [online] arXiv.org. Available from: https://arxiv.org/abs/1609.05473 [Accessed 27 Apr. 2024].
17. Fedus, W., Goodfellow, I., and Dai, A.M. 2018. *MaskGAN: Better Text Generation via Filling in the _____*. [online] arXiv.org. Available from: https://arxiv.org/abs/1801.07736 [Accessed 27 Apr. 2024].
18. Arjovsky, M., Chintala, S. and Bottou, L., 2017. *Wasserstein GAN*. [online] arXiv.org. Available from: https://arxiv.org/abs/1701.07875 [Accessed 29 Apr. 2024].

19. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V. and Courville, A., 2017. *Improved Training of Wasserstein GANs*. [online] arXiv.org. Available from: https://arxiv.org/abs/1704.00028 [Accessed 29 Apr. 2024].

20. Guo, J., Lu, S., Cai, H., Zhang, W., Yu, Y. and Wang, J., 2017. *Long Text Generation via Adversarial Training with Leaked Information*. [online] arXiv.org. Available from: https://arxiv.org/abs/1709.08624 [Accessed 28 Apr. 2024].

21. Marzoev, A., Madden, S., Kaashoek, M.F., Cafarella, M. and Andreas, J., 2020. *Unnatural Language Processing: Bridging the Gap Between Synthetic and Natural Language Data* [online] arXiv.org. Available from: https://arxiv.org/abs/2004.13645 [Accessed 22 Apr. 2024].

22. Wei, J., and Zou, K., 2019. *EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks*. [online] arXiv.org. Available from: https://arxiv.org/abs/1901.11196 [Accessed 29 Apr. 2024].

23. Li, Z., Zhu, H., Lu, Z., and Yin, M., 2023. *Synthetic Data Generation with Large Language Models for Text Classification: Potential and Limitations.* [online] arXiv.org. Available from: https://arxiv.org/abs/2310.07849 [Accessed 22 Apr. 2024].

24. Kumar, N., and Bojar, O., 2022. *Genre Transfer in NMT: Creating Synthetic Spoken Parallel Sentences using Written Parallel Data.* [online] ACL Anthology. Available from: https://aclanthology.org/2022.icon-main.28/ [Accessed 22 Apr. 2024].

25. Kim, T., Park, C., Hong, J., Dua, R., Choi, E., and Choo, J., 2022. *Reweighting Strategy based on Synthetic Data Identification for Sentence Similarity.* [online] ACL Anthology. Available from: https://aclanthology.org/2022.coling-1.429.pdf [Accessed 22 Apr. 2024].

26. Bayón, M. do C., and Sánchez-Gijón, P., 2019. *Evaluating Machine Translation in a Low-Resource Language Combination: Spanish-Galician.* [online] Papers With Code. Available from: https://paperswithcode.com/paper/evaluating-machine-translation-in-a-low [Accessed 22 Apr. 2024].