Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 4699–4708
Marseille, 11–16 May 2020

Exploring Transformer Text Generation for Medical Dataset Augmentation

Ali Amin-Nejad[1], Julia Ive[1], Sumithra Velupillai[2]
[1] Imperial College London, [2] King's College London
{ali.amin-nejad18, j.ive}@imperial.ac.uk, sumithra.velupillai@kcl.ac.uk

Abstract

Natural Language Processing (NLP) can help unlock the vast troves of unstructured data in clinical text and thus improve healthcare research. However, a big barrier to developments in this field is data access due to patient confidentiality which prohibits the sharing of this data, resulting in small, fragmented and sequestered openly available datasets. Since NLP model development requires large quantities of data, we aim to help side-step this roadblock by exploring the usage of Natural Language Generation in augmenting datasets such that they can be used for NLP model development on downstream clinically relevant tasks. We propose a methodology guiding the generation with structured patient information in a sequence-to-sequence manner. We experiment with state-of-the-art Transformer models and demonstrate that our augmented dataset is capable of beating our baselines on a downstream classification task. Finally, we also create a user interface and release the scripts to train generation models to stimulate further research in this area.

Keywords: Natural Language Generation, Language Modelling, Ethics and Legal Issues

1. Introduction

Natural Language Processing (NLP) has enormous potential to advance many aspects of healthcare by facilitating the analysis of unstructured text (Esteva et al., 2019). However a key obstacle to the development of more powerful NLP methods in the clinical domain is a lack of accessible data. This, coupled with the fact that state-of-the-art (SOTA) neural models are well known to require very large volumes of data in order to learn general and meaningful patterns, means that progress is hindered in this area. Data access is usually restricted due to the constraints on sharing personal medical information for confidentiality reasons, be they legal or ethical in nature (Chapman et al., 2011).

In the Machine Learning community, similar problems are typically solved by using artificially generated data to augment or perhaps even replace an original dataset (Bachman, 2016) in e.g. image processing. However, similar approaches to data augmentation are not easily applied to NLP. With language being inherently more complex than other domains, it is difficult to programmatically modify a sentence or document without altering the meaning and coherency. Natural Language Generation (NLG) can provide a more sophisticated approach to solving this problem and has already done so, e.g. in machine-translation with the technique known as *back-translation* (Sennrich et al., 2016). With newer, more capable, NLG models - utilising the Transformer architecture (Vaswani et al., 2017) - we posit that this general idea can now be extended beyond machine translation to longer passages of text.

Indeed NLG is an active area of NLP research, however there are still challenges to be addressed. The replacement or augmentation of genuine training data with artificial training data remains understudied, particularly in the medical domain. Attempting to achieve this manually, e.g. Suominen et al. (2015), is a costly and unscalable approach. Furthermore, the application of SOTA Transformer models for hierarchical generation beyond the sentence-level also remains understudied. Since most research focuses on shorter sentence-level texts, it is not clear whether these models can form sufficiently long range dependencies to be useful as a substitute for genuine training data. Therefore we believe that applying NLG approaches to medical text for augmentation purposes is a worthwhile research area in order to ascertain its viability. In the long term, if successful, we also aim to share this synthetic data with healthcare providers and researchers to promote methodological research and advance the SOTA, helping realise the potential NLP has to offer in the medical domain.

We build on the approaches of Liu (2018) and Melamud and Shivade (2019) in generating complex, hierarchical passages of text using a Transformer-based approach (Vaswani et al., 2017). We do this in both high-resource and low-resource scenarios to ensure that we assess the utility of NLG data augmentation in a low resource setting - when it is inherently most needed. We experiment with two Transformer architectures: the original vanilla architecture which achieved SOTA machine-translation results (Lakew et al., 2018), and the more recent GPT-2, composed of a stack of Transformer decoders, which has achieved SOTA question-answering, language modelling and common-sense reasoning results (Radford et al., 2019). We use this artificial data in two clinically relevant downstream NLP tasks (unplanned readmission prediction and phenotype classification) to effectively assess its utility both as a standalone dataset, and as part of an augmented dataset alongside the original samples. Our ultimate aim is to ascertain whether using SOTA Transformer models can generate new samples of text that are useful for data augmentation purposes - particularly in low resource medical scenarios.

Our main contributions are as follows: (i) we introduce a methodology to generate medical text for data augmentation; (ii) we demonstrate that our method shows promise by achieving significant results over our baselines on the readmission prediction task. This result is obtained using a pretrained BioBERT model.

We hope that this will pave the way for healthcare professionals in the field to appropriate this technique for the benefit of healthcare, stimulate further research, and enable the creation of entirely synthetic shareable clinical notes.

2. Related Work

Whilst NLG is an increasingly active area of NLP research, current SOTA approaches have not been extensively applied to the generation of medical text. Where it has, this has often been short excerpts of text as opposed to longer passages usually found in Electronic Health Records (EHRs) e.g. generation of imaging reports by Jing et al. (2018) or the generation of X-ray captions by Spinks and Moens (2018).

When it comes to the task of generating full EHRs, these EHRs often do not include the free text associated with the records (Choi et al., 2017), or the free text that is included is very short, such as the approach of Lee (2018) which generates chief complaints limited to 18 tokens or less. The closest published attempt to generate long passages of text in EHRs that we are aware of, is that of Liu (2018) who train a generative model using the public, de-identified MIMIC-III dataset (Johnson et al., 2016) and achieve reasonably coherent results on multiple measures, but do not perform any extrinsic evaluation to assess the quality of this text on downstream tasks. Another similar work is that of Melamud and Shivade (2019) who also utilise the MIMIC-III dataset to generate long passages of text and go further to study the utility of the artificial text in a number of downstream NLP tasks. However they do not use SOTA approaches for generation, opting for LSTMs over Transformers, and their downstream tasks are not clinically focused. Lastly, they do not study the utility of the synthetic text for augmentation purposes, only as a standalone dataset.

The closest overall approach to our own is that of Wang et al. (2019) who use the vanilla Transformer model to generate text and then evaluate using a phenotype classification task and a temporal evaluation task. Their text generation, however, is done at the sentence level before being joined together to form a full EHR note. This is unlike the approaches of Liu (2018) and Melamud and Shivade (2019) whose models output an entire EHR note in one iteration.

Finally, standard approaches to data augmentation are not easily applied to NLP due to the inherent complexity of language. Most approaches simply resort to randomly swapping words. This replacement can be done with synonyms, either by using a thesaurus (Zhang et al., 2015) or word embedding similarity (Wang and Yang, 2015), or it can be done with other entirely random words (Wang et al., 2018). A recent approach from Wei and Zou (2019) known as Easy Data Augmentation (EDA) combines some of these approaches and has been shown to improve performance for both convolutional and recurrent neural networks with particularly strong results for smaller datasets. We use this approach as a baseline for comparison against our NLG models for augmentation.

3. Methodology

3.1. Data

We use EHRs from the publicly available MIMIC-III database (Johnson et al., 2016), a large de-identified database for critical care hospital admissions at the Beth Israel Deaconess Medical Center, Boston MA. The version used for this research is the latest version (v1.4) which comprises over 58,000 hospital admissions for 38,645 adults and 7,875 neonates spanning June 2001 - October 2012. We are particularly concerned with the NOTEEVENTS table which comprehensively provides all the textual notes written by doctors, nurses and other healthcare professionals during a patient's stay. We focus solely on the Discharge Summaries, which provide the richest content about the patient's stay at the ICU.

The MIMIC-III database contains data only for neonates and adult patients (defined as being $>= 15$ years of age). For the purposes of this research, we remove the neonates due to the fact we believe there would be considerable and significant differences between the care of these two patients groups and this would be reflected in the discharge summaries for these patients. After removing these patients, we are left with 55,404 discharge summaries for 37,400 unique patients.

3.1.1. Dataset Split

We split our full dataset of 55,404 discharge summaries into training, validation and test datasets in the ratio 8:1:1. In the low-resource scenario where we experiment with an artificially smaller dataset, we keep the same validation set as the larger dataset and instead just shrink the size of the training and test datasets.

In order to determine the size of our low-resource dataset, we took inspiration from the recently introduced WikiText-2 and WikiText-103 datasets (Merity et al., 2016). These datasets are collated from Wikipedia entries and are often used to benchmark general-domain language models. They are named to reflect the number of words in each dataset with WikiText-2 containing $\sim$2m words and WikiText-103 containing $\sim$103m words. In accordance with this nomenclature, we name our low-resource and full-resource benchmarks, and henceforth refer to them as MimicText-9 and MimicText-98 respectively. Breakdowns of these datasets can be seen in Table 1.

In order to produce the training set for MimicText-9, we sample 4000 notes from the MimicText-98 training set. This results in $\sim$9m words in our training dataset, however, since Transformer models are limited to processing a maximum of 512 tokens and will truncate anything greater than this number, in practice gives us $\sim$2m words - the same as WikiText-2.

In order to produce the test set for MimicText-9, we reduce the size as much as we possibly can without affecting our downstream NLP tasks. The bottleneck in this case is the phenotype classification task where we need to predict the phenotypes for a curated dataset of $\sim$1600 admissions. Due to the fact that there are sometimes multiple ICU stays per admission, this corresponds to 1,846 discharge summaries. Reducing the size any more than this would adversely affect this downstream task so we leave the test set at this size.

3.2. Experimental Setup

We treat our text generation task as a conditional language modelling problem. More specifically, we model the task as a seq2seq problem where we generate discharge summaries conditioned on some input representing key information regarding the patient and their ICU stay following the approach of Liu (2018) generating each summary at the

	Train	Valid	Test
	MimicText-9		
Notes	4000	5,447	1,846
Words	9,048,735	12,187,184	4,446,003
	MimicText-98		
Notes	44,230	5,447	5,727
Words	98,243,403	12,187,184	13,332,263

	Train	Valid	Test
	WikiText-2		
Articles	600	60	60
Words	2,088,628	217,646	245,569
	WikiText-103		
Articles	28,475	60	60
Words	103,227,021	217,646	245,569

Table 1: Dataset comparison of MimicText vs WikiText

full note level. This leaves the attention mechanism of the model to entirely ascertain what portions of the input are relevant to what portions of the output. We believe that this is a viable approach given the advanced Transformer architecture we are using. In order to extract the relevant content from a patient's history, we explore the rest of the MIMIC-III dataset. Drawing on the approach of Liu (2018), we experiment with various configurations of the following context data classes in addition to a hint representing the first 10 tokens of the note. We settle on using all of the below data classes in the following order:

1. **Demographic data (G.A.E.):** This is static data which is found at the subject level. We extract gender and ethnicity, and compute the age at the time of the note using the date of birth of the patient and the date of the note.

2. **Diagnoses (D):** Intuitively, one can assume that diagnoses are a key element regarding a subject's stay in the ICU and would be extremely pertinent for writing the discharge summary. We include all International Classification of Diseases, Ninth Revision (ICD-9) codes for diagnoses pertaining to a patient's hospital admission ordered by priority, with the highest priority items first.

3. **Procedures (P):** Similar to diagnoses, procedures are also a key element of a subject's stay in the ICU. Again these are ICD-9 procedures but are instead ranked in the order in which they were performed.

4. **Medications (M):** Medications prescribed to the patient within a 24hr context window prior to discharge are included as context data. We include the name of the drug, the strength and the units.

5. **Microbiology Tests (T):** Nosocomial infections are those which are contracted during a hospital admission and and have a prevalence of 15% (Sydnor and Perl, 2011). We include the results of tests which test for these infections within a 72hr context window including the location of the test on the subject and the list of organisms detected at that location (if any).

6. **Laboratory Tests (L):** Lastly, we also include lab tests measuring normal bodily functions within a 24hr context window. We extract the name of the test, the value, its unit of measurement, and, if available, the flag saying whether or not this value is abnormal.

An example instantiation of this input context data is illustrated in Figure 1.

```
First ten tokens ... <H>
M <G>
65 <A>
white <E>
other pulmonary embolism and infarction | acute kidney failure ,
    unspecified | diarrhea | hypotension , unspecified <D>
other endoscopy of small intestine | gastroenterostomy without gastrectomy
    <P>
warfarin , 1mg Tablet | polysaccharide iron complex , 150MG | bisacodyl ,
    10MG SUPP | milk of magnesia , 30ML UDCUP <M>
blood culture : None | urine : staphylococcus species | mrsa screen : None
    | blood culture : None <T>
Calcium, Total , 10.0 , mg/dL | Bicarbonate , 25 , mEq/L | Hematocrit ,
    28.7 , \% , abnormal <L>
```

Figure 1: Example instantiation of input context for conditional generation

3.3. Pre-processing

We use the ScispaCy tokenizer (Neumann et al., 2019) to tokenize our text. ScispaCy is a specialised NLP library for processing biomedical texts which is built on top of the robust spaCy library[1]. We use the medium sized version of the library to tokenize our text: **en_core_sci_md**.
Additionally we convert all text to lower case and remove words which occur 3 times or fewer in our vocabulary. These out-of-vocabulary words are replaced with "$<$ UNK $>$". We also replace all newline characters with a new token "$<$ PAR $>$" representing a paragraph. This is due to the requirement that the input to our models must be in one line, however we do not wish to lose information regarding formatting of the note. Hence we replace it with a different token allowing us to recreate the formatting in post-processing.

3.4. Text Generation Models

As mentioned, we model the generation of text as a seq2seq problem. Whilst language models can be used standalone to generate text, we generally prefer to use conditional language models e.g. seq2seq. These usually consist of two architectures in an encoder-decoder format (Sutskever et al., 2014) where a source sequence is encoded into a latent space before being decoded to the target sequence. Transformers follow this paradigm having 6 encoder and 6 decoder layers, whilst GPT-2 instead only consists of Transformer decoder layers.
We use the vanilla Transformer implementation from the `tensor2tensor`[2] library (Vaswani et al., 2018) and we train each of our models for 3-4 epochs using a batch size of 4096 tokens and 4 Tesla K80 GPU chips, each having 12GB of RAM. We do this for both MimicText-9 and MimicText-98 using the 'transformer_base' hyperparameters provided

[1] www.spacy.io
[2] https://github.com/tensorflow/tensor2tensor

by tensor2tensor. We experiment with various combinations of input contexts using this vanilla Transformer to ascertain the optimal input context to use for the GPT-2 model and all downstream tasks. We decode using a beam size of 4 and alpha of 0.6 (default values).

We use the Tensorflow GPT-2 implementation directly from the OpenAI repository[3]. We fine-tune the pretrained "GPT-2 small" model (12 decoder layers) on both MimicText-98 and MimicText-9. We choose to focus only the small model for quicker and cheaper training due to its fewer parameters. We use the fine-tuning scripts provided by nshepperd[4]. We train the model for 60,000 steps using a batch size of 2 samples (2048 tokens) and 1 Tesla K80 GPU chip. We decode using a temperature of 1 and top-k of 40 (default values).

Since GPT-2 consists of only a stack of decoders, without any encoders, the problem needs modification for seq2seq tasks. Therefore, we follow the approach of the authors (Radford et al., 2019) who demonstrate that seq2seq tasks can be modelled by the introduction of a special token to help the model infer the desired task. We follow this framework and fine-tune the GPT-2 model using a context of examples pairs of the format `context data = target note` before conditioning the model with a prompt of `context data =` to generate `target note` at inference time.

3.5. Intrinsic Evaluation

The intrinsic evaluation step allows us to determine the shallow proximity of the generated text to the original. We report negative perplexity (neg. PPL), BLEU, ROUGE-2 and ROUGE-L. BLEU (Papineni et al., 2002) measures n-gram precision between generated and original text, while ROUGE-2 – bigram recall, and ROUGE-L – the longest in-sequence common n-gram recall. Negative PPL reflects the confidence of the model in the produced output (the higher the value the higher the confidence). For the BLEU score, we use the implementation provided by tensor2tensor, while for ROUGE we use `pyrouge`[5] - a python wrapper for the commonly used original ROUGE package from Lin (2004).

Tables 2 and 3 show our intrinsic evaluation results for models trained on MimicText-98 and MimicText-9 respectively. For both datasets, we train the vanilla Transformer model and GPT-2. We can see that the Transformer model outputs text significantly closer to the real text than GPT-2 on MimicText-98 whilst GPT-2 produces outputs closer to the real text than our Transformer model on MimicText-9.

Model	Neg. PPL	BLEU	ROUGE-2	ROUGE-L
Transformer	-2.117	4.76	0.3306	0.5942
GPT-2	-2.357	0.06	0.1350	0.1716

Table 2: Instrinsic Results for MimicText-98 test set

[3]https://github.com/openai/gpt-2

[4]https://github.com/nshepperd/gpt-2

[5]https://github.com/bheinzerling/pyrouge

Model	Neg. PPL	BLEU	ROUGE-2	ROUGE-L
Transformer	-2.474	0.10	0.3048	0.5699
GPT-2	-2.759	0.10	0.1309	0.1788

Table 3: Intrinsic Results for MimicText-9 test set

We hypothesise that for MimicText-98, there is enough data for the Transformer model to learn meaningful relationships and represent them in its output. Hence, the output is closer to the real text. For MimicText-9 however, there is not enough data for the Transformer model to learn these relationships, hence the considerably lower metrics across the board. For GPT-2 however, we see reasonably similar metrics for both MimicText-98 and MimicText-9. Since GPT-2 has been pre-trained on 40GB of internet text data, it has already learnt to model the English language and therefore needs considerably less data to achieve reasonable results and effectively learn how to write a discharge summary.

3.6. Downstream Tasks

3.6.1. Unplanned Readmission Prediction

For this task, we attempt to reproduce the work of Rajkomar et al. (2018) who perform a suite of various clinically relevant tasks such as mortality prediction, 30-day unplanned readmission, prolonged length of stay and final discharge diagnoses using EHR data from two hospitals in the US. Rajkomar et al. (2018) use the entire data from the EHR to do this whereas we focus solely on using the discharge summaries. The authors report the AUC scores 0.93-0.94, 0.75-76, 0.85-0.86 and 0.90 respectively for those tasks. Since our only data point is, by definition, at the end of the subject's stay, most of these tasks become either unfeasible or trivial. In our view, the only remaining relevant task is the 30-day unplanned readmission prediction, which is also incidentally the hardest task judging by the reported aforementioned AUC scores.

We label each discharge summary as either positive or negative depending on whether the patient then has a readmission within 30 days. Since we are only concerned with unplanned readmissions, as these the are the only ones where there is clinical value in predicting their occurrence, we filter for only the EMERGENCY and URGENT admission types (ignoring ELECTIVE and NEWBORN).

In order to ensure that our data for this task has not been seen by any of our text generation models before, we use the MimicText test sets to form the entire dataset for this task. We split the MimicText test sets for both MimicText-9 and MimicText-98 in the ratio 8:1:1 to form our training, validation and test sets for this task.

Samples where there is an unplanned readmission within 30 days only make up 6% of the discharge summaries in our entire dataset. In order to effectively deal with this imbalance, we ensure these instances are stratified across our training, validation and test sets. After splitting our dataset, we also then upsample our positive samples in the training set **only**, since heavy imbalance during training is well known to result in poor classification performance. To avoid losing any data, we opt to oversample our underrepre-

	Train	Valid	Test
	MimicText-9		
Original	2,412	185	185
Original 2x	4,824	185	185
Original EDA	4,824	185	185
Synthetic	2,412	185	185
Original + Synthetic	4,824	185	185
	MimicText-98		
Original	7,998	573	573
Original 2x	15,996	573	573
Original EDA	15,996	573	573
Synthetic	7,998	573	573
Original + Synthetic	15,996	573	573

Table 4: MimicText-9 and MimicText-98 dataset sizes for the Readmission Classification task

sented class (the readmissions) to be equal in number to the overrepresented class (non-readmissions). This then forms our primary baseline, which we call the 'Original' data. This upsampling is performed for all datasets in this task. Table 4 shows the final dataset sizes for all permutations.

We perform the classification using BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) and a variant of BERT, termed BioBERT (Lee et al., 2019). BERT is a recent Transformer-based architecture which has achieved SOTA results across numerous NLP tasks whilst BioBERT is a version of BERT pretrained on biomedical corpora demonstrating SOTA results (including significant improvements over BERT) on biomedical text mining tasks. We use BioBERT v1.1 (+ PubMed 1M) which takes an already pretrained BERT-base and trains it for a further 1M steps on the 4.5B word PubMed corpus[6].

We use the PyTorch (Paszke et al., 2017) implementation of BERT provided by the `pytorch-transformers` library[7] and train both BERT and BioBERT using the training scripts provided by the `fast-BERT` library[8], built on top of pytorch-transformers. We compare the performance of our synthetic data as input to the models both standalone and combined with the original data comparing against our 3 baselines: the original data, the original data augmented with a copy of itself (i.e. duplicated, mimicking a very bad generation model simply reproducing the original data without adding any variation to it) and the original data augmented with a copy of itself modified using the EDA technique described in Section 2. Therefore, we end up with 8 sets of results for each of our text-generation models and 12 sets of results for our baselines. We train the larger datasets for 6 epochs and the smaller datasets for 3 epochs.

3.6.2. Phenotype Classification

Our phenotype classification task is borrowed from Gehrmann et al. (2018) and is the same task conducted by Wang et al. (2019). We model this as a multilabel classification task where subjects are categorised as demonstrating up to 13 different phenotypes ranging from the likes of Obesity and Alcohol Abuse to Advanced Cancer and Depression. The dataset is a carefully curated subset of 1610 discharge summaries from MIMIC-III with the annotations made by a panel of medical professionals.

Although the dataset is a subset of MIMIC-III, the authors initially actually used MIMIC-II to collect the data. Due to some structural differences between MIMIC-III and MIMIC-II, we could not identify the exact discharge summaries for hospital admissions where there was > 1 discharge summary per admission (i.e. multiple ICU stays). In these cases, we kept all the discharge summaries pertaining to an admission on the assumption that the subjects will be exhibiting the same phenotypes for all the ICU stays in a given admission. This resulted in an increase of the dataset size from 1,610 to 1,846. This was then split in the ratio 8:1:1 to form our training, validation and test sets respectively. Due to the large number of different classes, and the relatively small size of the dataset, it was infeasible to ensure stratification of classes, so the splits were simply performed randomly.

Again, we perform the classification using the BERT and BioBERT models and for each of our text generation models, we compare the performance of our synthetic data as input to the models both standalone and combined with the original data for our models trained on MimicText-98. As mentioned, since we have a curated dataset for this task, we do not explore the low-resource scenario and therefore text trained on MimicText-9 is not used in this task. We compare the results against our 3 baselines: the original data, the original data augmented with a copy of itself (i.e. duplicated) and the original data augmented with a copy of itself modified using the EDA technique. We do this for both BERT and BioBERT models training the larger datasets for 10 epochs and the smaller datasets for 20 epochs. Therefore, we end up with 8 sets of results for each of our text-generation models and 6 sets of results for our baselines.

4. Results

4.1. Readmission Prediction

For the Readmission Prediction task, we report Accuracy, AUC and F1 scores for both our BERT and BioBERT classification models. Additionally, due to the aforementioned difficulty of this task, we also include the metrics recall and precision in order to obtain some more granular information regarding the performance of our models. Table 5 shows our results.

Looking at MimicText-98, our first observation is the strong performance of the BioBERT model for the 'Combined' data generated from our Transformer model. This beats the 'Original' baseline on most metrics, including ones that matter most in classification tasks - AUC, recall and F1. And indeed in 2/3 of these metrics, the results are significant at the 90% or 95% confidence level. These results also highlight the flaws of accuracy as a metric for an imbalanced classification problem. This is particularly evident from the two most accurate models which have precision, recall and F1 values of exactly zero, showing they did not even predict a single positive sample correctly, which also highlights the difficulty of this task.

[6]https://www.ncbi.nlm.nih.gov/pubmed/

[7]https://github.com/huggingface/pytorch-transformers

[8]https://github.com/kaushaltrivedi/fast-bert

	Dataset	Model	MimicText-98					MimicText-9				
			AUC	Acc.	Prec.	Recall	F1	AUC	Acc.	Prec.	Recall	F1
Base-line	Original	BioBERT	0.6060	0.8691	**0.4615***	0.1644	0.2424	0.4673	0.7838	**0.3125**	0.1471	0.2000
		BERT	0.5914	0.8621	0.3500	0.0959	0.1505	0.5027	0.5405	0.1600	0.3529	0.2202
	Original 2x	BioBERT	0.6115	0.8325	0.2195	0.1233	0.1579	0.5319	0.7622	0.2222	0.1176	0.1538
		BERT	0.5775	0.8447	0.2143	0.0822	0.1188	0.4901	0.6811	0.0968	0.0882	0.0923
	Original EDA	BioBERT	0.6142	0.8464	0.2727	0.1233	0.1698	**0.6428**	0.7730	0.3000	0.1765	0.2222
		BERT	0.4404	**0.8726**	0.0000	0.0000	0.0000	0.4965	**0.8162**	0.0000	0.0000	0.0000
Trans-former	Synthetic	BioBERT	0.5206	0.8674	0.2000	0.0137	0.0256	0.5273	**0.8162**	0.0000	0.0000	0.0000
		BERT	0.4479	0.8464	0.1053	0.0274	0.0435	0.5045	**0.8162**	0.0000	0.0000	0.0000
	Combined	BioBERT	**0.6690**	0.8534	0.3878	**0.2603****	**0.3115***	0.5275	0.6162	0.1967	0.3529	0.2526
		BERT	0.5356	0.8569	0.2632	0.0685	0.1087	0.4599	**0.8162**	0.0000	0.0000	0.0000
GPT-2	Synthetic	BioBERT	0.4803	0.8709	0.3333	0.0137	0.0263	0.5860	0.5892	0.2162	**0.4706***	**0.2963**
		BERT	0.4878	0.8551	0.1429	0.0274	0.0460	0.5164	**0.8162**	0.0000	0.0000	0.0000
	Combined	BioBERT	0.5807	0.8447	0.2500	0.1096	0.1524	0.5223	0.7243	0.2703	0.2941	0.2817
		BERT	0.5163	**0.8726**	0.0000	0.0000	0.0000	0.5228	**0.8162**	0.0000	0.0000	0.0000

Table 5: Readmission prediction results for MimicText-98 and MimicText-9. Results for the best model in each category are highlighted in bold. * = significance at the 90% confidence level. ** = significance at the 95% confidence level

These results show promise for using the Transformer model to augment our original dataset. However, interestingly, this benefit only manifests itself when classifying with the BioBERT model. We see significant improvements for most datasets at the 95% confidence level for using BioBERT over BERT, but we did not anticipate this sheer increase in improvement for the Transformer augmented dataset. In fact, the BERT version of this model considerably underperforms the BERT model for the 'Original' dataset, and yet the BioBERT model significantly outperforms the 'Original' BioBERT model as well as all other baselines and our GPT-2 models.

We hypothesise that this significant improvement is due to the fact that the synthetic data adds just the optimal amount of noise to our original dataset, allowing our BioBERT model to learn more general relationships and avoid overfitting the training dataset. Crucially however, we believe it is the pretraining undertaken by BioBERT on biomedical data which allows it to do this, as evident by the lacklustre performance demonstrated by BERT.

For MimicText-9, as expected, our baseline and Transformer models perform significantly worse due to the significantly smaller dataset. Our GPT-2 model is the best performer demonstrating significant recall values at the 90% confidence level for the synthetic text. We hypothesise this performance is due to the extensive pretraining of the GPT-2 model. Furthermore, as expected, all our BioBERT models outperform our BERT models across the board. Overall, it appears that using the Transformer model for augmentation in the low resource scenario is not a viable option. We cannot say for certain whether using GPT-2 or EDA could positively impact our results. However, it appears that our EDA baseline generally performs even worse than our Transformer and GPT-2 augmentations and the Original data itself for both MimicText-98 and MimicText-9.

Overall, we believe that our synthetic text could have useful implications for augmenting datasets to improve performance on downstream clinically relevant tasks. Performing a qualitative evaluation on our samples, whilst our samples generally resemble a standard discharge summary with a relatively consistent narrative, there are often inaccuracies that are even apparent to the untrained eye, e.g. references to broken hips for a fall patient despite an x-ray of the area being mentioned as unremarkable. It is our hypothesis that these inaccuracies can provide an optimal amount of noise when using a model that has been pretrained on biomedical texts, thus allowing them to better generalise. However, this noise proves too much for models that have only been pretrained on non-medical text.

4.2. Phenotype Classification

Due to the sheer volume of results for each of our 13 phenotypes, we only report a summary in this section detailing accuracy, AUC and F1 scores. Table 6 shows our average results across the 13 phenotypes for MimicText-98. In accordance with the literature, we report a macro-weighted AUC and a micro-weighted F1 for our average results using the implementations from `scikit-learn`[9].

Our first observation is that our results are all very similar across all our metrics. Indeed we can not identify a model which is significantly better than all the rest at the 95% or even 90% confidence level for any of our metrics. This leads us to hypothesise that this task might be too easy and that even weaker models are able to relatively accurately identify the phenotypes of patients from their discharge summaries. However, we still note that our baseline models report the highest values across our metrics, especially our 'Original' data using the BioBERT model which reports the best accuracy and F1 scores.

In addition, we also cannot conclusively say that the 'Original' data performs better than when it is augmented with our Transformer or GPT-2 models ('Original + Synthetic') at the 95% confidence level when comparing the same model (i.e. only comparing BERT with BERT and BioBERT with BioBERT). What we can conclusively say however is that the 'Original + Synthetic' datasets significantly outperform our 'Synthetic' datasets for both Trans-

[9]https://scikit-learn.org

	Dataset	Model	Acc.	AUC	F1
Baseline	Original	BioBERT	**0.9005**	0.7737	**0.4863**
		BERT	0.8864	0.7632	0.4326
	Original 2x	BioBERT	0.8921	**0.7743**	0.4760
		BERT	0.8841	0.7578	0.3984
	Original EDA	BioBERT	0.8932	0.7491	0.3797
		BERT	0.8841	0.7497	0.4278
Transformer	Synthetic	BioBERT	0.8754	0.6680	0.1183
		BERT	0.8716	0.6638	0.0865
	Combined	BioBERT	0.8887	0.7693	0.3909
		BERT	0.8860	0.7588	0.3450
GPT-2	Synthetic	BioBERT	0.88716	0.5818	0.1289
		BERT	0.8704	0.5961	0.0907
	Combined	BioBERT	0.8906	0.7398	0.3571
		BERT	0.8868	0.7486	0.3318

Table 6: Phenotype Classification results for MimicText-98 showing average accuracy, AUC and F1 scores. Bold results indicate the best model for that metric.

former and GPT-2 models. Our final observation is that BioBERT generally outperforms BERT across most metrics, regardless of the dataset - reinforcing the results from our readmission prediction task. We do not include our Transformer and GPT-2 results for MimicText-9. This is due to the fact that we are using a curated dataset of the same size and therefore this comparison between MimicText-9 and MimicText-98 is not useful in this instance.

We posit that this task is more sensitive to the exact inaccuracies in our synthetic text compared to the readmission prediction task. Since we are predicting phenotypes, our models will be attending heavily to the actual words of these phenotypes and other closely related words - which do not necessarily always reflect the true phenotypes of the patient. For example, one sample has alternating references to both hypertension and hypotension in the same note. Logically, this would impact our model predictions for 'Advanced Heart Disease' and possibly other phenotypes. Our results indicate this noise does not lead to better generalisation and simply reduces performance. However, our readmission prediction task is a more complex problem which can manifest itself in the text in myriad more abstract ways. It appears that our models can represent these manifestations sufficiently capably, despite minor inaccuracies, so as to boost overall performance on the task.

5. Discussion

5.1. Contributions

We show that the vanilla Transformer architecture is able to adequately learn a hierarchical, long passage seq2seq task when trained on a large enough dataset (MimicText-98). Whilst this artificial text is by itself of poorer quality than the original text, we find that when it is used to augment our original dataset, it can boost results on downstream tasks, specifically the readmission prediction task. We hypothesise this is due to it introducing sufficient noise that allows the models to avoid overfitting and generalise better. Furthermore, where this is not the case, it still achieves results comparable to that of the Original dataset and other augmentation baselines.

Conversely, we show that the vanilla Transformer architecture is not able to do this on a small dataset (MimicText-9). The quality of this artificial text is too poor which results in lower performance on our downstream evaluation tasks. This indicates that it is not always be suitable for data augmentation in low-resource scenarios. The GPT-2 model on the other hand shows promising results in these cases achieving comparable results with our baselines. We believe this is due to the extensive pretraining it has undergone meaning it needs less data to learn this transformation. However, we also show that the GPT-2 model is not ideal for long passage seq2seq tasks as demonstrated by its lower performance compared to the Transformer model on MimicText-98 and the fact that there is very little difference between its models trained on MimicText-98 and MimicText-9. Our results suggest that this is due to the fact that it does not rely on the encoder-decoder paradigm like the Transformer and is better suited to conditional language modelling as opposed to seq2seq tasks.

Lastly, we demonstrate the utility of BioBERT as an alternative to BERT for certain clinical NLP tasks. For our datasets, we show that it almost always achieves better results than BERT. Indeed, our results in the readmission prediction task also indicate that a classification model pretrained on biomedical data such as BioBERT may not only be helpful, but possibly also necessary to harness the statistical power of our synthetic data.

5.2. Models and Interface

We also create a user interface and intend to release the trained models for the research community in order to help stimulate further research in this area. The models have been submitted for release on the PhysioNet[10] website. Our interface is displayed in Figure 2. The left side is where the user can input the patient information discussed in Section 3.2, whilst the right hand side depicts a blurred out version of the output Discharge Summary. This is done for confidentiality reasons and adherence to the MIMIC-III terms of use. Instructions to reproduce our models and the code for this interface are provided on GitHub[11].

5.3. Applications

In addition to bettering clinical outcomes, an important possible application of this research is its potential to reduce the significant time spent by clinicians writing clinical notes, as also highlighted by Liu (2018). According to a study by Sinsky et al. (2016), physicians spend almost two hours of their time doing administrative work for every hour of time spent with patients. The bulk of this time-consuming administrative work is the inputting of clinical notes into EHR software detailing patient history, assessment, treatment plan, etc. – all of which are in the discharge summaries we have been focusing on. This imposes a significant burden on clinicians and healthcare providers as a whole, resulting in an array of downstream effects such as

[10] www.physionet.org

[11] https://github.com/amin-nejad/mimic-text-generation; https://github.com/amin-nejad/mimic-website

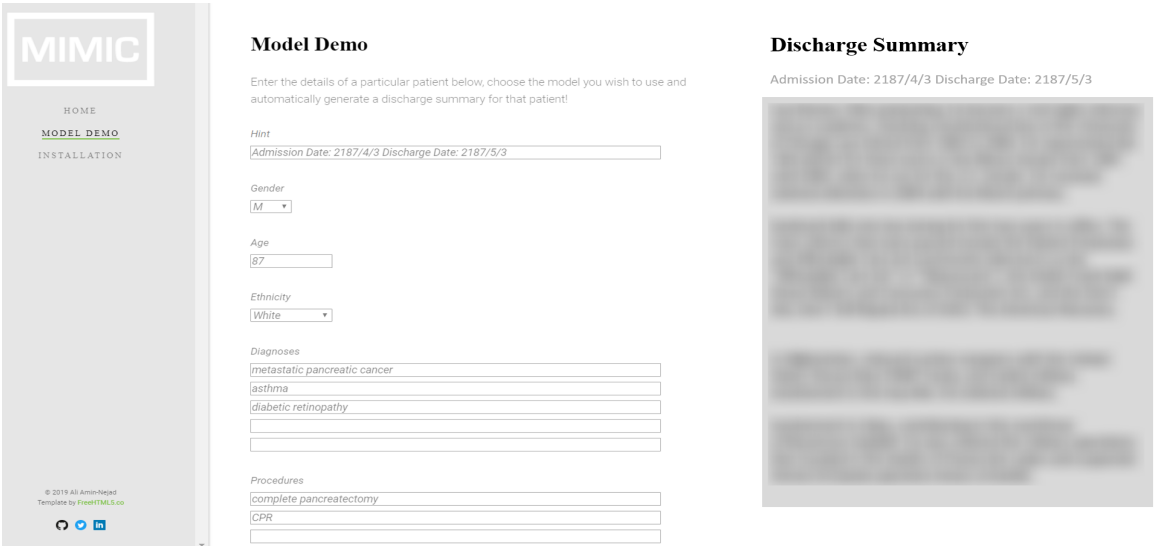

Figure 2: Model interface

longer patient waiting times, overworked clinicians and ultimately a misallocation of resources. We aspire that this work will also contribute to the goal of NLP-assisted note-writing by clinicians, thus saving time, money and improving both clinician and patient satisfaction (Friedberg et al., 2014; Shanafelt et al., 2016).

5.4. Future Work

We only experiment with the vanilla Transformer model and GPT-2-small. However, we would like to see this research extended to larger architectures, particularly ones which have shown superior performance in modelling long range dependencies such as the Transformer with Memory Compressed Attentioned (Liu et al., 2018) and the recently introduced Transformer-XL (Dai et al., 2019). In addition, we would also like to explore further pretraining. The GPT-2 model we trained showed promise in the low-resource scenario due to its heavy pretraining but ultimately fell short overall due to its lack of encoders. We would like to continue our experiments with pretrained transformer models, thus combining the two best qualities of the Transformer and GPT-2 models with which we experimented. This would be particularly promising, if we also ensured the pretraining was done on biomedical data in a similar fashion to BioBERT. We posit that a *BioTransformer* would do the same for language generation on biomedical tasks such as producing discharge summaries.

A big barrier to advancing NLP in healthcare is maintaining the privacy of patients. Whilst this is in fact the very motivation for our research, it is also susceptible to its effects. We are still not at a stage yet where we can release de-identified patient data publicly, whether this genuine or synthetic, due to their unquantified susceptibility to re-identification attacks. Therefore, even if we develop a model which shows strong performance in generating synthetic clinical notes, these notes cannot be easily shared with the wider research community. We welcome more research in this field that incorporates susceptibility to re-identification attacks.

6. Conclusion

In this paper, we have explored the use of SOTA Transformer models for the purposes of medical text augmentation. Specifically, we focused on the vanilla Transformer and GPT-2 models to generate discharge summaries from the MIMIC-III dataset, modelled as a seq2seq task. We initially explore different ways to represent our input context data before moving on to evaluate our output discharge summaries. We assess the quality of this synthetic data both standalone and augmented with the original data on two downstream clinically relevant NLP tasks: readmission prediction and phenotype classification. We compare our results against the original data as well as a more conventional data augmentation baseline using word replacement known as EDA.

Our results show that whilst the synthetic data is generally of poorer quality, it can yield results significantly better than our baselines on the readmission prediction task. Crucially, we also show that these results only manifest themselves when using the BioBERT model, which has been pretrained on biomedical documents. Our research demonstrates some promising results but further work is needed in this area to ascertain the viability of this approach to medical data augmentation.

7. Acknowledgements

The work of Julia Ive is part-funded by EPSRC Healtex Feasibility Funding (Towards Shareable Data in Clinical Natural Language Processing: Generating Synthetic Electronic Health Records). Sumithra Velupillai is part-funded by the National Institute for Health Research (NIHR) Biomedical Research Centre at South London and Maudsley NHS Foundation Trust and King's College London, and by the Medical Research Council (MRC) Mental Health Data Pathfinder Award to Kingâs College London. The views expressed are those of the authors and not necessarily those of the NHS, the NIHR or the Department of Health and Social Care.

8. Bibliographical References

Bachman, P. (2016). An architecture for deep, hierarchical generative models. In *Advances in Neural Information Processing Systems*, pages 4826–4834.

Chapman, W. W., Nadkarni, P. M., Hirschman, L., D'Avolio, L. W., Savova, G. K., and Uzuner, O. (2011). Overcoming barriers to nlp for clinical text: the role of shared tasks and the need for additional creative solutions. *Journal of the American Medical Informatics Association*, 18(5):540–543.

Choi, E., Biswal, S., Malin, B., Duke, J., Stewart, W. F., and Sun, J. (2017). Generating multi-label discrete patient records using generative adversarial networks. In Finale Doshi-Velez, et al., editors, *Proceedings of the 2nd Machine Learning for Healthcare Conference*, volume 68 of *Proceedings of Machine Learning Research*, pages 286–305, Boston, Massachusetts, 18–19 Aug. PMLR.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q., and Salakhutdinov, R. (2019). Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy, July. Association for Computational Linguistics.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., Cui, C., Corrado, G., Thrun, S., and Dean, J. (2019). A guide to deep learning in healthcare. *Nature medicine*, 25(1):24.

Friedberg, M. W., Chen, P. G., Van Busum, K. R., Aunon, F., Pham, C., Caloyeras, J., Mattke, S., Pitchforth, E., Quigley, D. D., Brook, R. H., et al. (2014). Factors affecting physician professional satisfaction and their implications for patient care, health systems, and health policy. *Rand health quarterly*, 3(4).

Gehrmann, S., Dernoncourt, F., Li, Y., Carlson, E. T., Wu, J. T., Welt, J., Foote Jr, J., Moseley, E. T., Grant, D. W., Tyler, P. D., et al. (2018). Comparing deep learning and concept extraction based methods for patient phenotyping from clinical narratives. *PloS one*, 13(2):e0192360.

Jing, B., Xie, P., and Xing, E. (2018). On the automatic generation of medical imaging reports. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2577–2586, Melbourne, Australia, July. Association for Computational Linguistics.

Johnson, A. E., Pollard, T. J., Shen, L., Li-wei, H. L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., and Mark, R. G. (2016). Mimic-iii, a freely accessible critical care database. *Scientific data*, 3:160035.

Lakew, S. M., Cettolo, M., and Federico, M. (2018). A comparison of transformer and recurrent neural networks on multilingual neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 641–652, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2019). Biobert: a pre-trained biomedical language representation model for biomedical text mining. *arXiv preprint arXiv:1901.08746*.

Lee, S. H. (2018). Natural language generation for electronic health records. *NPJ digital medicine*, 1(1):63.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.

Liu, P. J., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, L., and Shazeer, N. (2018). Generating wikipedia by summarizing long sequences. *CoRR*, abs/1801.10198.

Liu, P. J. (2018). Learning to write notes in electronic health records. *arXiv preprint arXiv:1808.02622*.

Melamud, O. and Shivade, C. (2019). Towards automatic generation of shareable synthetic clinical notes using neural language models. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 35–45.

Merity, S., Xiong, C., Bradbury, J., and Socher, R. (2016). Pointer sentinel mixture models. *CoRR*, abs/1609.07843.

Neumann, M., King, D., Beltagy, I., and Ammar, W. (2019). Scispacy: Fast and robust models for biomedical natural language processing.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners.

Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., Liu, P. J., Liu, X., Marcus, J., Sun, M., et al. (2018). Scalable and accurate deep learning with electronic health records. *NPJ Digital Medicine*, 1(1):18.

Sennrich, R., Haddow, B., and Birch, A. (2016). Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany, August. Association for Computational Linguistics.

Shanafelt, T. D., Dyrbye, L. N., Sinsky, C., Hasan, O., Satele, D., Sloan, J., and West, C. P. (2016). Relationship between clerical burden and characteristics of the electronic environment with physician burnout and professional satisfaction. In *Mayo Clinic Proceedings*, volume 91, pages 836–848. Elsevier.

Sinsky, C., Colligan, L., Li, L., Prgomet, M., Reynolds,

S., Goeders, L., Westbrook, J., Tutty, M., and Blike, G. (2016). Allocation of physician time in ambulatory practice: a time and motion study in 4 specialties. *Annals of internal medicine*, 165(11):753–760.

Spinks, G. and Moens, M.-F. (2018). Generating continuous representations of medical texts. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 66–70, New Orleans, Louisiana, June. Association for Computational Linguistics.

Suominen, H., Zhou, L., Hanlen, L., and Ferraro, G. (2015). Benchmarking clinical speech recognition and information extraction: new data, methods, and evaluations. *JMIR medical informatics*, 3(2):e19.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Sydnor, E. R. and Perl, T. M. (2011). Hospital epidemiology and infection control in acute-care settings. *Clinical microbiology reviews*, 24(1):141–173.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Vaswani, A., Bengio, S., Brevdo, E., Chollet, F., Gomez, A. N., Gouws, S., Jones, L., Kaiser, L., Kalchbrenner, N., Parmar, N., Sepassi, R., Shazeer, N., and Uszkoreit, J. (2018). Tensor2tensor for neural machine translation. *CoRR*, abs/1803.07416.

Wang, W. Y. and Yang, D. (2015). That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using# petpeeve tweets. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2557–2563.

Wang, X., Pham, H., Dai, Z., and Neubig, G. (2018). SwitchOut: an efficient data augmentation algorithm for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 856–861, Brussels, Belgium, October-November. Association for Computational Linguistics.

Wang, Z., Ive, J., Velupillai, S., and Specia, L. (2019). Is artificial data useful for biomedical natural language processing algorithms? In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 240–249.

Wei, J. W. and Zou, K. (2019). Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.