# 1 Notes on Reproducibility

## 1.1 Total Computational Budget and Infrastructure used

For *ZESHEL* task, we had to train BLINK's bi-encoder with BERT-base pre-trained model and GAN-LM model for augmentation. To do so, we have employed `p3.8xlarge` AWS instances. Each instance has 4x NVIDIA V100 GPUs. Each approach reported in the paper was trained three times with different seeds with each run being allocated their own GPU. Training times varied based on setting: For full training set, runs it took up to 9 hours to train a BLINK's bi-encoder and GAN-LM model. For low-resource, runs it took up to 1.5 hours.

For *TREC, STS-B* and *mSTS* tasks, we used NVIDIA Tesla T4 to train the downstream models (i.e. BERT-Tiny, SentenceTransformer, mean pooling of mBERT) and GAN-LM approach. We also trained three times with different seeds for getting the averaged results. For *TREC*, it required up to 18 minutes for low-resource and 1.16 hours for half-train set scenario while for *STS-B*, it took up to 2 minutes for low-resource and 12 minutes for half-train set. For *mSTS*, it needed 2 minutes for low-resource, and 8 minutes for normal scenario.

## 1.2 Hyperparameters and Size Estimation in Downstream Models

For all downstream tasks, we mostly followed the default parameters found in the original repositories (see Table 1). The number of parameters in *ZESHEL, TREC, STS-B, mSTS* are approximately 220M, 4.4M, 125M, 179M respectively.

## 1.3 GAN-LM

For GAN-LM, we chose to run training for 10 epochs in the case of *ZESHEL* and 50 epochs in the cases of *TREC, STS-B, mSTS* as we've noticed that both the discriminator and the generator converge well for these settings. For fine-tuning in *mSTS*, additional 10 epochs for each language is applied. `dimension` was chosen to be 27 for *ZESHEL*, 25 for *TREC*, 36 for *STS-B*, and 24 for *mSTS* to cover the 99% of data in each databases. We employ the Adam optimizer with $lr$ = 0.0001, $\beta_1$ = 0, $\beta_2$ = 0.9 in both generator and discriminator, and trained the discriminator 5 times more than the generator to balance both networks. As GAN-LM uses BART-base and mBART-large-50 as its encoder and decoder, the overall model is around 139M and 610M parameters with up to 14M parameters added by GAN-LM as reported in Table 2. In addition, we show the example of input data for GAN-LM in Figure 1 where each token of text is stacked as the dimension and zero values are added for remaining parts.

Table 1: Hyperparameters in downstream tasks.

| Parameter | Setting |
|---|---|
| ***ZESHEL*** | |
| learning_rate | 1e-5 |
| num_train_epochs | 10 |
| max_context_length | 128 |
| max_cand_length | 128 |
| train_batch_size | 32 |
| eval_batch_size | 32 |
| bert_model | bert-base-uncased |
| type_optimization | all_encoder_layers |
| ***TREC*** | |
| learning_rate | 2e-5 |
| num_train_epochs | 60 |
| train_batch_size | 32 |
| pre_trained_model | bert-tiny |
| warmup_proportion | 0.1 |
| seq_len | 128 |
| ***STS-B*** | |
| num_train_epochs | 4 |
| train_batch_size | 16 |
| pre_trained_model | xlm-roberta-base |
| learning_rate_scheduler | warmuplinear |
| warmup_proportion | 0.1 |
| evaluation_steps | 1000 |
| ***mSTS*** | |
| num_train_epochs | 10 |
| train_batch_size | 64 |
| pre_trained_model | bert-base-multilingual-cased |
| learning_rate_scheduler | warmuplinear |
| warmup_proportion | 0.1 |
| evaluation_steps | 1000 |

# 2 Databases

In Table 3, we show the summary of each dataset with its downstream task and in Figure 2, we cover the distribution of data regarding classes in *TREC* dataset which confirms the unbalance of data.

# 3 Additional Ablation Study and Augmented Examples

Table 4 reveals the architectural ablation study for GAN-LM and Tables 5, 6 shows the additional examples of augmented data.

Table 2: GAN-LM structure. (a) Generator. (b) Discriminator. Conv, Conv-T and FC mean the convolutional, convolutional-transpose and fully connected layers. Batch normalization (BN), LeakyReLu and dropout are applied after each operation. To train GAN, the sum between the noise from Gaussian distribution ($\mu = 0$, $\sigma^2 = 1$) and the input embedding from text is considered. The output from generator is same size as the input embedding to build the synthetic texts from dictionary in pre-trained LM. In (b), the discriminator gives a single value as output. In *mSTS*, we consider the average pooling in discriminator, instead of stride in convolutional layers.

**(a)**

| Layer | Operation | Kernel Size (W x H x D) | Stride | BN, LeakyReLu |
|---|---|---|---|---|
| **1** | FC | - | - | Yes |
| **2** | Conv-T #1 | 5 x 1 x 128 | 2 | |
| **3** | Conv-T #2 | 5 x 1 x 64 | 3 (4: *mSTS*) | |
| **4** | Conv-T #3 | 5 x 1 x 32 | 2 | |
| **5** | Conv-T #4 | 5 x 1 x # of tokens | 2 | No |

**(b)**

| Layer | Operation | Kernel Size (W x H x D) | Stride | LeakyReLu, Dropout (50%) |
|---|---|---|---|---|
| **1** | Conv #1 | 5 x 1 x 32 | 2 (1: *mSTS*) | Yes |
| **2** | Conv #2 | 5 x 1 x 64 | 2 (1: *mSTS*) | |
| **3** | Conv #3 | 5 x 1 x 128 | 3 (1: *mSTS*) | |
| **4** | Conv #4 | 5 x 1 x 256 | 2 (1: *mSTS*) | |
| **5** | FC | - | - | No |

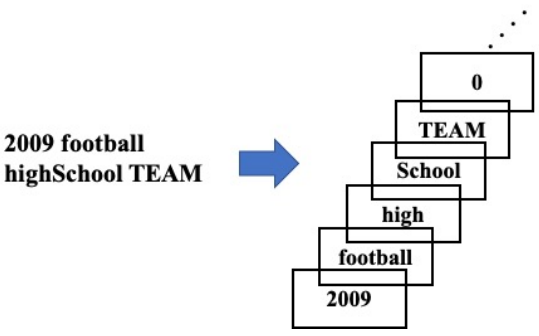


Figure 1: The description of input data in GAN-LM. Each token is assigned in each dimension and the rest of parts is filled with zero values.

Table 3: Details of datasets used.

| Datasets | Division | Size | Downstream |
|---|---|---|---|
| *ZESHEL* | Train | 49k | Entity Linking |
| | Validation | 10k | |
| | Test | 10k | |
| *TREC* | Train | 5k | Question Classification |
| | Test | 500 | |
| *STS-B* | Train | 5k | Semantic Textual Similarity |
| | Validation | 1k | |
| | Test | 1k | |
| *mSTS* | Train | 2k | Multilingual Semantic Textual Similarity |
| | Test | 1k | |

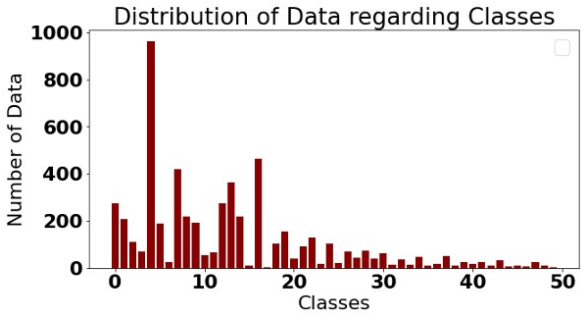


Figure 2: The distribution of data according to classes in *TREC* which confirms that it is highly unbalanced.

Table 4: GAN-LM study in *STS-B* with half-train set. Original structure of GAN-LM is detailed in Table 2.

| Type | SRC |
|---|---|
| GAN-LM with BART (0.3 - 0.7) * | **78.02%** |
| GAN-LM with BART (0.1 - 0.5) | 75.57% |
| GAN-LM with BART (0.5 - 0.9) | 77.49% |
| GAN-LM with BERT (0.3 - 0.7) | 71.33% |
| GAN-LM with XLNet (0.3 - 0.7) | 74.21% |
| Change Convolutional to Dense layer in * | 75.99% |
| Change LeakyReLU to ReLU in * | 75.66% |
| Eliminate Batch Normalization in * | 76.32% |
| Eliminate Dropout in * | 75.89% |
| Use AveragePooling in * | 76.29% |

## 4 Links for considered Datasets and Models

- *ZESHEL*: https://github.com/lajanugen/zeshel. CC-BY-SA License.
- *TREC*: https://cogcomp.seas.upenn.edu/Data/QA/QC/.
- *STS-B*: https://ixa2.si.ehu.eus/stswiki/index.php/STSbenchmark.
- *mSTS*: https://alt.qcri.org/semeval2017/task1/, https://www.sbert.net/examples/training/multilingual/README.html.
- BLINK: https://github.com/facebookresearch/BLINK. MIT License.
- BERT-Tiny, BERT-base, mBERT-BASE: https://github.com/google-research/bert. Apache-2.0 License.

Table 5: Additional examples of generated augmentations. Bold texts in each cell denote the changed parts.

| Type | Example |
|---|---|
| Original | The man is riding a motorcycle down the road. |
| Lexical | The **mankind** is riding a motorcycle down the road. |
| Spelling | The man is **rideing** a motorcycle down the road. |
| Character | The man is **giding** a **motorcymle** down the road. |
| Token-LM | **I** man is riding a **SUV** down the road. |
| Back-Translation | The man is riding a motorcycle **along the way.** |
| Paraphrase | The man **rides down the road a motorcycle.** |
| OPT | The man is riding a motorcycle down the road. **There's not actually any other cars there** |
| GPT | The man is riding a motorcycle down the road. **He begins to look over his shoulder** |
| GAN-LM | **A** man is riding a **SUV** down the road. |
| GAN-LM-GPT | **A** man is riding a **SUV** down the road. **He begins to look over his shoulder** |

Table 6: Examples of generated augmentations in multilingual case. In each sentence, we include the result of Google Translator. For GAN-LM-Back, back-translation is considered in Arabic language and GAN-LM is applied in other languages.

| Type | Example |
|---|---|
| Original (AR) | علاخلا يف شيجلا ضرعلا كانهو - There is an army parade in the open |
| GAN-LM | علاخلا يف يركسع ضرع كانه - There is a military parade outside |
| GAN-LM-Back | قلطلا ءاوهلا يف شيجلا ضرعلا وه اهو - And here is the outdoor army parade |
| Original (ES) | La mujer está cuidando al niño. - The woman is taking care of the child. |
| GAN-LM | la mujeres estaba cuidando A niños. - the women were taking care of children. |
| Original (DE) | Er war in der Nähe einer Brücke. - He was near a bridge. |
| GAN-LM | Sie waren in der Nähe einer Brücke! - They were near a bridge! |
| Original (TR) | Bu küçük kız düğün için heyecanlı - This little girl is excited for the wedding |
| GAN-LM | bu büyük kız evlilik İçin heyecan lılar - this big girl is excited for marriage |
| Original (FR) | Quatre femmes sont dans la piscine. - Four women are in the pool. |
| GAN-LM | Quatre filles étaient dans la piscine. - Four girls were in the pool. |
| Original (IT) | L'uomo è entusiasta, poiché questa è la sua prima presentazione per la sua nuova azienda - The man is thrilled, as this is his first presentation for his new company |
| GAN-LM | maschio è fanata, poiché questa è la Sua prima presentación per la suas nuova il lavoro - male is fanatic, since this is his first presentation for his new work |
| Original (NL) | Drie mensen dragen moderne kleding. - Three people wear modern clothes. |
| GAN-LM | Drie mensen dragen moderne meubels. - Three people carry modern furniture. |

- SentenceTransformers: https://www.sbert.net. Apache-2.0 License.
- GPT-2: https://github.com/openai/gpt-2. Modified MIT License.
- mGPT: https://github.com/THUNLP-MT/PLM4MT. BSD-3-Clause License
- OPT: https://github.com/facebookresearch/metaseq. MIT License.
- T5: https://github.com/google-research/text-to-text-transfer-transformer. Apache-2.0 License
- Prism: https://github.com/thompsonb/prism. MIT License.
- Helsinki-NLP/Opus-MT: https://github.com/Helsinki-NLP/Opus-MT. MIT License.
- BART-base: https://github.com/facebookresearch/fairseq/tree/main/examples/bart. MIT License.
- mBART-large-50: https://huggingface.co/facebook/mbart-large-50. MIT License.
- RoBERTa-large: https://github.com/facebookresearch/fairseq/blob/main/examples/roberta/README.md. MIT License.
- XLNet-base: https://github.com/zihangdai/xlnet. Apache-2.0 License.