Supplementary Material: How Resilient are Language Models to Text Perturbations?

Daniel Romero-Alvarado, José Hernández-Orallo, and Fernando Martínez-Plumed

Universitat Politècnica de València
dromalv@vrain.upv.es, jorallo@upv.es, fmartinez@dsic.upv.es

A Model details

For reproducibility and a deeper understanding, this section provides detailed configurations of the hyperparameters used for those transformer-based models adapted for sequence classification tasks. The models detailed here include:

- `DistilBERT` [6] (Table 1), a streamlined version of BERT optimised for lower resource consumption through knowledge distillation: a larger "teacher" model trains a smaller "student" model.
- `ELECTRA` [2] (Table 2), another BERT-based model that is resource-efficient and potentially more effective than BERT and XLNet. It uses a generator to replace tokens with alternatives and a discriminator to identify these changes during pre-training.
- `Funnel Transformer` (Table 3) [3], which integrates pooling operations to reduce layer size and includes an upsampling layer to achieve specific sequence lengths.
- `XLNet` [10] (Table 4), which builds on Transformer-XL [4] and extends bidirectional learning by permuting input tokens while preserving token dependencies, requiring more computational resources.

Each table lists critical parameters such as model type, architecture, vocabulary size, embedding dimensions, dropout rates, and specific settings tailored to enable these models to efficiently classify sequences.

Table 1: Hyperparameters: `DistilBERT` for sequence classification.

Parameter	Value
_name_or_path	distilbert-base-cased
activation	gelu
architectures	DistilBertForSequenceClassification
attention_dropout	0.1
dim	768
dropout	0.1
hidden_dim	3072
initializer_range	0.02
max_possition_embeddings	512
model_type	distilbert
n_heads	12
n_layers	6
output_past	true
pad_toke_id	0
problem_type	single_label_classification
qa_dropout	0.1
seq_classif_dropout	0.2
sinusoidal_pos_embs	false
tie_wieights	true
torch_dtype	float32
transformers_version	4.29.2
vocab_size	28996

Table 2: Hyperparameters: `ELECTRA` for sequence classification.

Parameter	Value
_name_or_path	google/electra-base-discriminator
architectures	ELECTRAForSequenceClassification
attention_probs_dropout_prob	0.1
classifier_dropout	NULL
embedding_size	768
hidden_act	gelu
hidden_dropout_prob	0.1
hidden_size	768
initializer_range	0.02
intermediate_size	3072
layer_norm_eps	1.00E-12
max_position_embeddings	512
model_type	electra
num_attention_heads	12
num_hidden_layers	12
pad_token_id	0
position_embedding_type	absolute
problem_type	single_label_classification
summary_activation	gelu
summary_last_dropout	0.1
summary_type	first
summary_use_proj	true
torch_dtype	float32
transformers_version	4.29.2
type_vocab_size	2
use_cache	true
vocab_size	30522

B Dataset details

Here we provide a comprehensive overview of the five domains used in our experiments:

- **Sentiment analysis**: Using the Stanford Sentiment Treebank (SST2) [7], this task assesses binary sentiment (positive or negative) from film review excerpts.

Table 3: Hyperparameters: `Funnel Transformer` for sequence classification.

Parameter	Value
pame_or_path	funnet-transformer/small
activation dropout	ˆ
architectures	FunnelForSequenceClassification
attention dropout	0.1
attention_type	relative_shift
block_repeats	3 blocks of 1
block_sizes	3 blocks of size 4
d_head	64
d_inner	3072
d_model	768
hidden_act	pelu_new
hidden_dropout	0.1
initializer_range	0.1
initializer_std	NULL
layer$_{n}orm_{e}sp$	1.0E-9
max_position_embeddings	512
model_type	funnel
n_head	*2
num_decoder_layers	2
pooling_only	true
pooling_type	mean
problem_type	single_label_classification
rel_attn_type	factorised
separate_cls	true
torch_dtype	float32
transformers_version	4.29.2
truncate_seq	true
type_vocab_size	3
vocab_size	30522

- **Grammatical Acceptability**: The Corpus of Linguistic Acceptability (CoLA) [8] contains sentences from linguistic publications and tests binary grammatical correctness.
- **Semantic Similarity**: Using the Paraphrase Adversaries from Word Scrambling (PAWS) dataset [11], this task involves evaluating the binary similarity between sentence pairs generated by word scrambling.
- **Natural Language Inference**: Using the Multi-Genre Natural Language Inference Corpus (MNLI) [9], this task distinguishes between entailment, contradiction, or neutrality in sentence pairs from different genres.
- **Hate speech and offensive language**: Using the Hate Speech and Offensive Language Dataset (HSOL) [5], this task categorises tweets as hate speech, offensive language, or neither.

We detail the number of samples per split, subsampling requirements (for efficiency and compute reasons) and class distributions for each dataset. These datasets vary widely in size and composition, reflecting the diverse nature of tasks and challenges in NLP. Tables 6, 7, 8, 9 and 10 show illustrative examples for each dataset.

Table 4: Hyperparameters: `XLNet` for sequence classification.

Parameter	Value
_name_or_path	xlnet-base-casedd
architectures	XLNetForSequenceClassification
attn_type	bi
bi_data	false
bos_token_id	1
clamp_len	-1
d_head	64
d_inner	3072
d_model	768
dropout	0.1
end_n_top	5.00E+00
eos_token_id	2
ff_activation	gelu
initializer_range	0.02
layer_norm_eps	1.00E-12
model_type	xlnet
mem_len	NULL
n_head	12
n_layer	12
pad_token_id	5
problem_type	single_label_classification
reuse_len	NULL
same_length	false
start_n_top	5
summary_activation	tanh
summary_last_dropout	0.1
summary_type	last
summary_use_proj	true
torch_dtype	float32
transformers_version	4.29.2
untie_r	true
use_mems_eval	true
use_mem_train	false
vocab_size	32000

Table 5: Summary of datasets and their characteristics

Dataset	Split	# of Samples	Subsampling	Class Distribution
CoLA	Train	8851	No	30% Class 0, 70% Class 1
	Validation	1043	No	30% Class 0, 70% Class 1
	Test	1063	No	Labels unknown
HSOL	Train	24783	Yes	6% Class 0, 77% Class 1, 17% Class 2
MNLI	Train	392702	Yes	33% Class 0, 33% Class 1, 33% Class 2
	Validation	9815	No	35% Class 0, 32% Class 1, 33% Class 2
	Test	9796	No	Labels unknown
SST2	Train	67349	Yes	44% Class 0, 56% Class 1
	Validation	827	No	49% Class 0, 51% Class 1
	Test	1281	No	Labels unknown
PAWS	Train	49401	Yes	56% Class 0, 44% Class 1
	Validation	8000	No	56% Class 0, 44% Class 1
	Test	8000	No	56% Class 0, 44% Class 1

C Hyperparameter optimisation

All training processes used a batch size of 32, except XLNet sessions, which, due to the limited size of the GPU, used a batch size of 16. Table 11 shows the hyperparameters used in the finetunning phase.

Table 6: Examples for the Stanford Sentiment Treebank (STT2) with their corresponding label

Sentence	**Label**
cold movie	0 (negative)
with his usual intelligence and subtlety	1 (positive)
will find little of interest in this film, which is often preachy and poorly acted	0 (negative)
a $ 40 million version of a game	0 (negative)
gorgeous and deceptively minimalist	1 (positive)

Table 7: Examples for the Corpus of Linguistic Acceptability (CoLA) with their corresponding label

Sentence	**Label**
As you eat the most, you want the least	0 (not acceptable)
John was lots more obnoxious than Fred	1 (acceptable)
The tube was escaped by gas	0 (not acceptable)
We want John to win	1 (acceptable)
We persuaded Mary to leave and Sue to stay	1 (acceptable)

Table 8: Examples of the Paraphrase Adversaries from Word Scrambling (PAWS) dataset with their corresponding label

Sentence 1	**Sentence 2**	**Label**
It is the seat of Zerendi District in Akmola Region	It is the seat of the district of Zerendi in Akmola region	1 (Semantically similar)
BA relocated the former BCal routes to Tokyo and Saudi Arabia to Heathrow	BA transferred the former BCal routes to Heathrow to Tokyo and Saudi Arabia	0 (Not semantically similar)
He is trained by Daniel Jacobs and shares a gym with former world champion Andre Rozier	He is trained by Andre Rozier and shares a gym with the former World Champion Daniel Jacobs	0 (Not semantically similar)
The Leurda River is a tributary of the River Tabaci in Romania	The Tabaci River is a tributary of the River Leurda in Romania	0 (Not semantically similar)
Steam can also be used, and does not need to be pumped	Also steam can be used and need not be pumped	1 (Semantically similar)

Table 12 illustrates the results of the fine-tuning phase. Each model-task pair was evaluated with different hyperparameter combinations to determine their respective scores. The hyperparameter combination that produced the highest score was selected as the final model for the evaluation phase. In this table we show the results for all models and tasks, including the best number of epochs (left) and the optimal initial learning rate (right) for each pairing.

Table 9: Examples of the Multi-Genre Natural Language Inference Corpus (MNLI) dataset with their corresponding label

Premise	Hypothesis	Label
Gays and lesbians	Heterosexuals	2 (Contradiction)
yeah i mean just when uh the they military paid for her education	The military didn't pay for her education	2 (Contradiction)
(Read for Slate's take on Jackson's findings)	Slate had an opinion on Jackson's findings	1 (Neutral)
yeah well you're a student right	Well you're a mechanics student right?	O (Entailment)
Well you're a mechanics student right?	yeah well you're a student right	O (Entailment)

Table 10: Examples for the Hate Speech and Offensive Language (HSOL) dataset with their corresponding label

Sentence	Label
@rhythmixx_: hobbies include: fighting Mariam bitch	1 (Offensive language)
@AllAboutManFeet: http://t.co/3gzUpfuMev woof woof and hot soles	2 (Neither)
@CB_Baby24: @white_thunduh alsarabsss hes a beaner smh you can tell hes a mexican	0 (Hate Speech)
@CauseWereGuys: Going back to school sucks more dick than the hoes who attend it	1 (Offensive language)
@ArizonasFinest6: Why the eggplant emoji doe? y he say she looked like scream Imao	2 (Neither)

Several notable trends emerge from the results. The initial learning rate of 0.00005 generally produced the best results, while 0.0001 was never the optimal choice and sometimes produced vanishing or exploding gradients. It is also worth noting that the Grammatical Coherence task had the same number of training epochs for all models, highlighting its greater complexity compared to the other tasks. In summary, the results vary depending on the hyperparameters, task and model. However, some trade-offs can be found. For instance, an initial learning rate between 0.00005 and 0.00001 seems to be the best value, since learning rates higher than that, such as 0.0001, lead to worse results overall.

D Hardware, reproducibility and reporting results

Both finetuning and evaluation were performed in a server with an Intel® Core™ i9- 10920X at 3.5 GHz, a 126 GB memory, and 2 NVIDIA GeForce RTX 3090 GPUs with 24GB of dedicated memory each. The perturbations were made in CPU while the finetuning and prediction were performed in GPU.

To ensure full reproducibility, and in line with the guidelines recommended by a Science paper on AI evaluation reporting [1], all fine-tuning and evaluation

Table 11: Hyperparameters employed in the finetuning phase. The only two who take various values are the initial learning rate and the number of epochs

Hyperparameter	**Value**
Batch size	32 (16 XLNet)
Initial learning rate	0.0001, 0.0005/0.00001
Learning rate decay	Linear decay per epoch
# of epochs	5/7/10
Optimizer	Adam with weight decay (AdamW)
Weight decay	0.01
Adam β_1	0.9
Adam β_2	0.999
Adam ϵ	1×10^{-8}

Table 12: Performance of various models on different NLP tasks, including detailed epochs and learning rates

Model	**COLA**		**HSOL**		**MNLI**		**SST2**		**PAWS**	
	Epochs	*LR*	*Epochs*	*LR*	*Epochs*	*LR*	*Epochs*	*LR*	*Epochs*	*LR*
DistilBERT	10	0.00005	7	0.00005	7	0.00005	10	0.00005	5	0.00005
ELECTRA	10	0.00005	10	0.00005	5	0.00005	7	0.00005	5	0.00005
Funnel Transformer	10	0.00001	10	0.00001	10	0.00005	7	0.00001	5	0.00005
XLNet	7	0.00001	10	0.00001	5	0.00005	5	0.00005	5	0.00001

data is accessible via the provided repository[1] to avoid recomputation and therefore avoidable energy consumption. Despite the fixing of various seeds related to `transformers`, `PyTorch` and `NumPy` to control randomness in the fine-tuning phase, certain GPU processes remain non-deterministic, which may cause slight variations in the reproduced results. However, these variations should not affect the overall conclusions.

References

1. Burnell, R., Schellaert, W., Burden, J., Ullman, T.D., Martinez-Plumed, F., Tenenbaum, J.B., Rutar, D., Cheke, L.G., Sohl-Dickstein, J., Mitchell, M., et al.: Rethink reporting of evaluation results in ai. Science **380**(6641), 136–138 (2023)
2. Clark, K., Luong, M.T., Le, Q.V., Manning, C.D.: Electra: Pre-training text encoders as discriminators rather than generators. In: International Conf. on Learning Representations (2020)
3. Dai, Z., Lai, G., Yang, Y., Le, Q.V.: Funnel-transformer: Filtering out sequential redundancy for efficient language processing. In: NeurIPS (2020)
4. Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q.V., Salakhutdinov, R.: Transformer-xl: Attentive language models beyond a fixed-length context. Proc. of the 57th Annual Meeting of the Association for Computational Linguistics (2019)
5. Davidson, T., Warmsley, D., Macy, M., Weber, I.: Automated hate speech detection and the problem of offensive language. In: Eleventh International AAAI Conf. on Web and Social Media (2017)

[1] https://github.com/Daniframe/TFG-GCD

6. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In: NeurIPS EMC2 Workshop (2019)
7. Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. Proc. of the 2013 Conf. on empirical methods in NLP pp. 1631–1642 (2013)
8. Warstadt, A., Singh, A., Bowman, S.R.: Neural network acceptability judgments. Transactions of the Association for Computational Linguistics **6**, 625–641 (2018)
9. Williams, A., Nangia, N., Bowman, S.R.: A broad-coverage challenge corpus for sentence understanding through inference. In: Proc. of the 2018 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). pp. 1112–1122 (2018)
10. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R.R., Le, Q.V.: Xlnet: Generalized autoregressive pretraining for language understanding. NeurIPS **32** (2019)
11. Zhang, Y., Baldridge, J., He, L.: Paws: Paraphrase adversaries from word scrambling. In: Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. vol. 1, pp. 1298–1308 (2019)