

Light and Fast Language Models for Spanish Through Compression Techniques

José Cañete López

Supervisor: Felipe Bravo-Marquez
Department of Computer Science
Universidad de Chile

April 4, 2023

Overview

1. Motivation
2. Problem
3. Hypothesis and Objectives
4. Background and Related Work
5. Preliminaries: Evaluation Tasks and Baselines
6. Proposed Spanish NLP Resources: **ALBETO** and **Speedy Gonzales**
7. Results and Discussion
8. Conclusions

Motivation

Motivation

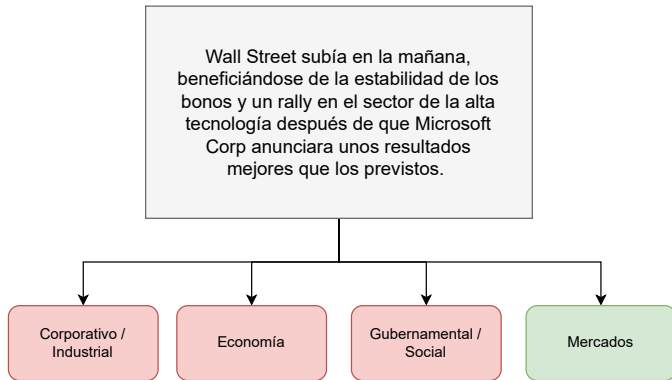



Figure: An example of document classification. Taken from the MLDoc [42] dataset.

Motivation

En su lugar entró el chileno Iván Luis Zamorano



PER

Figure: An example of named entity recognition. Taken from the CoNLL2002 NER [46] dataset.

Motivation

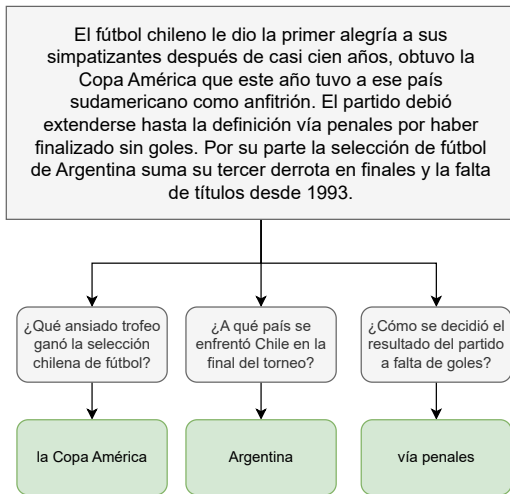
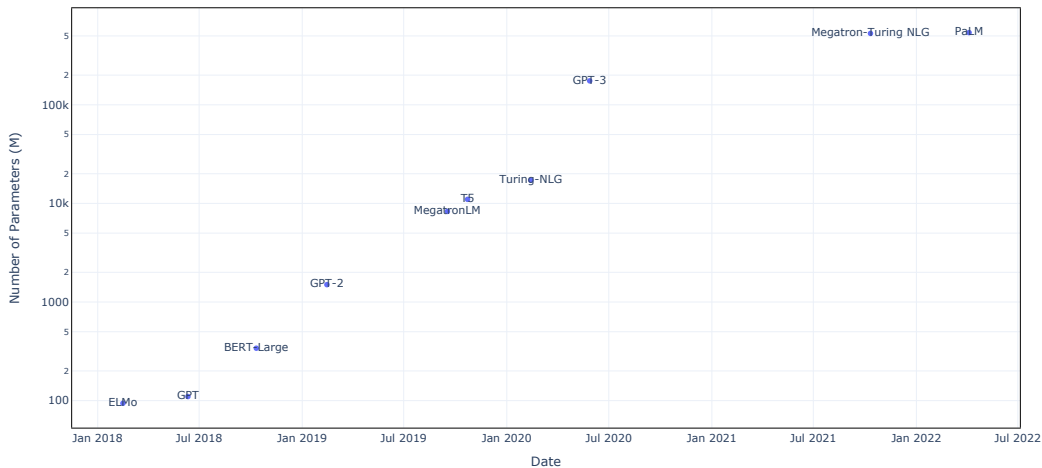


Figure: An example of question answering. Taken from the SQAC [20] dataset.

Motivation



Problem

Problem

As **models grows** in size and computational complexity, it's **difficult** to put them in **production** for real time applications or the use of them in hardware restricted devices like mobile phones.

Problem

And even **more difficult** for the **Spanish** language because of the lack of Spanish-specific resources and models.

Hypothesis and Objectives

Hypothesis

Adopting more parameter-efficient model architectures and employing knowledge distillation techniques to transfer knowledge from larger models to smaller ones can significantly enhance model compactness and inference speed, while maintaining most of the performance exhibited by larger models on Spanish NLP tasks.

General Objective

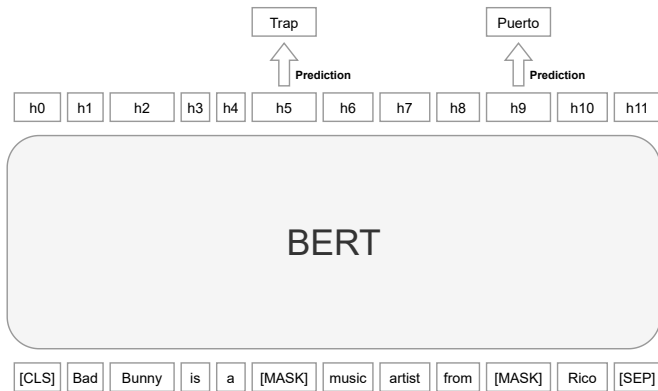
To develop **Spanish language models** that are more **compact** and **computationally efficient** while maintaining high levels of task-performance.

Specific Objectives

1. To measure the size and inference speed of pre-trained Spanish language models that are currently available.
2. To develop models for the Spanish language that are more parameter-efficient by utilizing weight-shared model architectures.
3. To train models for Spanish that are more inference-efficient by applying task-specific knowledge distillation on Spanish NLP tasks.
4. To evaluate the mentioned techniques on a diverse set of Spanish NLP tasks.
5. To evaluate how the model size impacts the task performance while using these techniques.
6. To release those models publicly as a resource for further research.

Background and Related Work

Background - BERT



- Bidirectional Encoder Representations from Transformers [17].
- Transformer-encoder.
- Pre-training on MLM and NSP.
- Fine-tuning on downstream tasks.
- *base* (110M) and *large* (330M).

Figure: The masked language modeling (MLM) task used by BERT as pre-training task.

Background - ALBERT

- A Lite BERT [24].
- Transformer-encoder.
- Embedding factorization and Parameter-sharing.
- Pre-training on MLM and SOP.
- Fine-tuning on downstream tasks.
- *base* (12M) to *xxlarge* (235M).

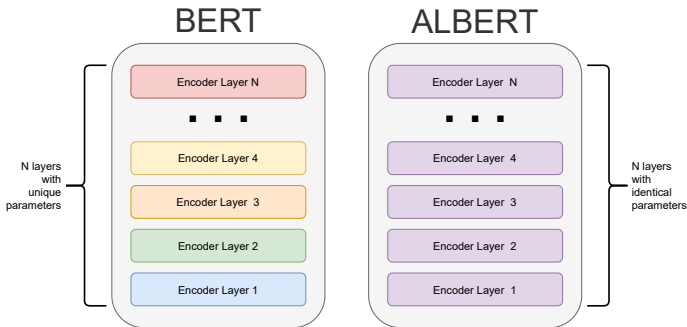


Figure: A design comparison of BERT and ALBERT, focusing on the parameter utilization strategy adopted by each model.

Background - Multilingual and Monolingual Models

- Multilingual models:
 - Models that are trained simultaneously using data from several languages.
 - Examples: mBERT (104 languages), XLM-R (100 languages).
 - Generally, larger vocabularies, to be able to represent all languages.
- Monolingual models:
 - Models trained on a single language.
 - CamemBERT [29] and FlauBERT [25] for French, BERTje [14] and RobBERT [15] for Dutch, FinBERT [49] for Finish, BETO [10] for Spanish.
 - Generally outperform multilingual models.

Background - Compression Techniques

- Methods to reduce the overall size or computational complexity of a model.
- **Pruning:** aims to reduce the number of connections (weights) in a neural network by identifying and removing redundant connections.
- **Quantization:** compresses the original network by reducing the number of bits required to represent each weight.
- **Knowledge Distillation:** transfers the knowledge from a big model (teacher) to a smaller model (student) by training the student to imitate the teacher.

Background - Knowledge Distillation

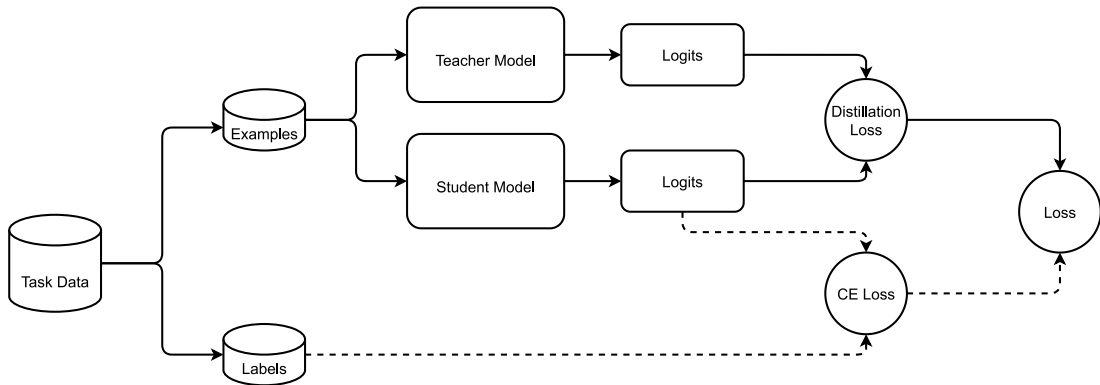


Figure: The figure provides a visual representation of the Knowledge Distillation [21] framework applied in this work.

Background - Knowledge Distillation

Two models, the teacher model, say M_T , and a student model, say M_S . We train M_S to imitate M_T . We define the distillation objective as L_{KD} :

$$L_{KD} = L_O(M_T(x), M_S(x))$$

Where L_O is a loss function that works on the logits of M_T and M_S . The most common choices for this loss are the cross entropy loss, the KL-divergence loss and the mean-squared error loss.

Also, we can include the gold labels from the training dataset. The complete loss, accounting these labels can be seen as:

$$L = \alpha L_{CE} + (1 - \alpha) L_{KD}$$

Where L_{CE} is the traditional cross-entropy loss against gold labels and $\alpha \in [0, 1]$ defines the weight of each loss.

Related Work

- Tang et al. [44] uses KD to transfer the knowledge from BERT to lighter RNNs.
- Turc et al. [47] proposes pre-training compact BERT models and then using task-specific KD to achieve better results.
- Sanh et al. [41] introduces a task-agnostic scheme where KD is used on the pre-training task.
- Wang et al. [50] and Jiao et al. [22] proposed different methods exclusive for Transformers, to directly distill the knowledge from the self-attention layers of the teacher model to the student model.

Preliminaries: Evaluation Tasks and Baselines

Evaluation Tasks

1. Text Classification

- Document Classification.
- Natural Language Inference.
- Paraphrase Identification.

2. Sequence Tagging

- Named Entity Recognition.
- Part-of-Speech Tagging.

3. Question Answering

Dataset Name	Task Type	Number of Categories	Train Size	Validation Size	Test Size
MLDoc [42]	Text Classification	4	9458	1000	4000
PAWS-X [52]	Text Classification	2	49401	2000	2000
XNLI [11]	Text Classification	3	392702	2490	5010
POS [45]	Sequence Tagging	18	14305	1654	1721
NER [46]	Sequence Tagging	9	8324	1916	1518
MLQA [27]	Question Answering	-	81810	500	5253
SQAC [20]	Question Answering	-	15036	1864	1910
TAR / XQuAD [6, 2]	Question Answering	-	87595	10570	1190

Table: Details of the datasets used to evaluate our proposed models.

Inference Metrics

Metrics

- Size: Number of Parameters.
- Speed: Multiply-accumulate Operations (MACs).

Conditions

- Batch size = 1.
- Max. sequence length = 512.

Pre-trained Models for Spanish

Aim

- Include all publicly available Transformer-encoder based models trained on Spanish general domain corpora as baselines.

Model Name	Architecture	Size	Vocab Size	Vocab Types	Max Seq Length	Parameters	Domain	Availability	Reference
Included									
BETO	BERT	base	32K	uncased, cased	512	110M	General	Public	[10]
DistilBETO	DistilBERT	base	32K	uncased	512	67M	General	Public	[18]
RoBERTa-BNE base	RoBERTa	base	50K	cased	514	125M	General	Public	[20]
RoBERTa-BNE large	RoBERTa	large	50K	cased	514	355M	General	Public	[20]
BERTIN	RoBERTa	base	50K	cased	514	125M	General	Public	[12]
Not Included									
GPT-2-BNE base	GPT-2	base	50K	cased	512	124M	General	Public	[20]
GPT-2-BNE large	GPT-2	large	50K	cased	512	773M	General	Public	[20]
RigoBERTa	DeBERTa	base	50K	-	512	-	General	Private	[43]
RoBERTuito	RoBERTa	base	30K	uncased, cased, deaccented	130	109M	Social Media	Public	[34]
BSC-Bio	RoBERTa	base	50K	cased	514	125M	Biomedical	Public	[7]
RoBERTalex	RoBERTa	base	52K	cased	514	126M	Legal	Public	[19]
Longformer-BNE	Longformer	base	50K	cased	4098	149M	General	Public	-

Table: Summary of pre-trained Transformer models for Spanish.

Proposed Spanish NLP Resources: ALBETO and Speedy Gonzales

ALBETO: Light models for Spanish

ALBETO: a series of 5 lightweight models that follow the ALBERT architecture and are pre-trained exclusively on Spanish corpora with sizes that range from 5M to 223M of parameters.

ALBETO: Model Architecture

- ALBERT architecture.
- 31K lowercase subword tokens.

Model	Parameters	Layers	Hidden	Embedding
ALBETO <i>tiny</i>	5M	4	312	128
ALBETO <i>base</i>	12M	12	768	128
ALBETO <i>large</i>	18M	24	1024	128
ALBETO <i>xlarge</i>	59M	24	2048	128
ALBETO <i>xxlarge</i>	223M	12	4096	128

Table: The configurations of each ALBETO model trained in this work.

ALBETO: Training Process

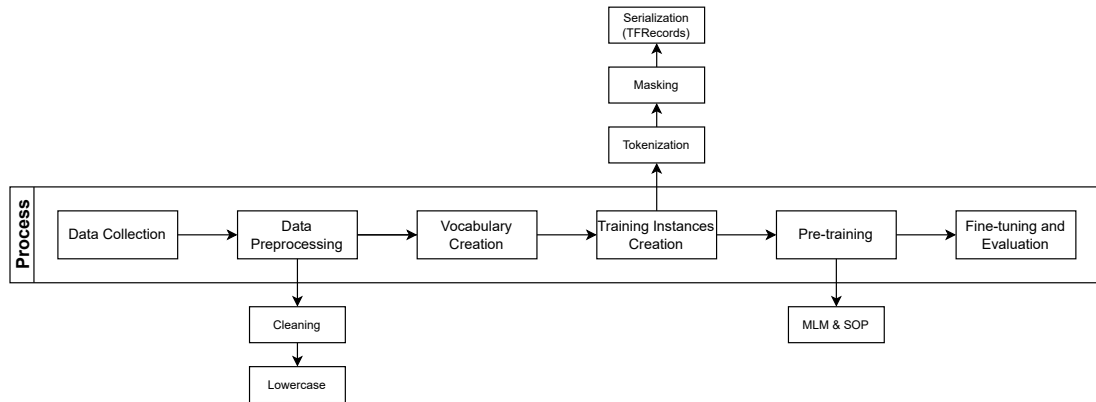


Figure: A broad overview of the process involved in the creation of ALBETO models. Sub-processes relevant to distinct stages are portrayed outside the main frame.

ALBETO: Evaluation

- Fine-tuning on downstream tasks.
- Tasks: text classification, sequence tagging and question answering.
- Hyperparameter search:
 - All models:
 - Batch size: 16, 32, 64.
 - Epochs: 2, 3, 4.
 - BETO, DistilBETO, RoBERTa-BNE, BERTIN, ALBETO *tiny* and *base*:
 - Learning rate: 1e-5, 2e-5, 3e-5, 5e-5.
 - ALBETO *large*, *xlarge* and *xxlarge*:
 - Learning rate: 1e-6, 2e-6, 3e-6, 5e-6.

Speedy Gonzales: Fast Models for Spanish

Speedy Gonzales: a collection of fast task-specific language models based on ALBETO, which were trained using Task-specific Knowledge Distillation.

Speedy Gonzales: Approach

Candidate Teacher Models

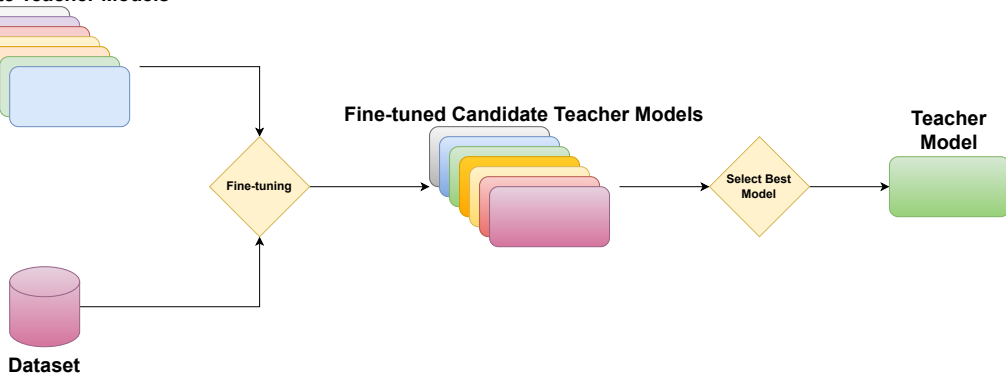


Figure: The first stage of our approach, which involves fine-tuning a set of candidate models on a specific dataset, followed by the selection of the best-performing model as the teacher model for that dataset.

Speedy Gonzales: Approach

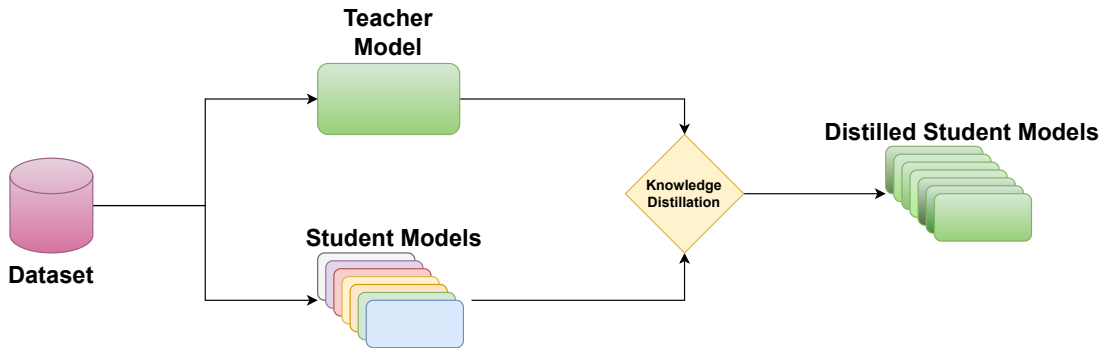


Figure: The second stage of our approach, which employs the selected teacher model to train a set of student models using knowledge distillation.

Speedy Gonzales: Approach

Candidate teacher models:

- All publicly available Transformer-encoder based models trained on Spanish general domain corpora.
- BETO, DistilBETO, RoBERTa-BNE, BERTIN and ALBETO.

Student models:

- ALBETO *tiny*.
- A collection of faster models based on ALBETO *base*:
 - Models that follows the ALBETO *base* configuration, but with less layers.
 - Noted as ALBETO *base-n*, $n \in (2, 4, 6, 8, 10)$.

Speedy Gonzales: Evaluation

First stage:

- Same as ALBETO evaluation.
- Selected best teacher models.

Second stage:

- Task-specific Knowledge Distillation on downstream tasks.
- Tasks: text classification, sequence tagging and question answering.
- KL-Divergence loss, $\alpha = 0$ and $T = 1$.
- Cached teacher predictions.
- Hyperparameter search:
 - Batch size: 16, 32, 64.
 - Learning rate: 5e-5, 1e-4.
 - Epochs: 50.
 - Early stopping with tolerance of 10 epochs of no improving.

Results and Discussion

Task Performance - Text Classification

Model	MLDoc	PAWS-X	XNLI
Fine-tuning			
BETO uncased	96.38	84.25	77.76
BETO cased	96.65	89.80	81.98
DistilBETO	96.35	75.80	76.59
ALBETO tiny	95.82	80.20	73.43
ALBETO base	96.07	87.95	79.88
ALBETO large	92.22	86.05	78.94
ALBETO xlarge	95.70	89.05	81.68
ALBETO xxlarge	96.85	89.85	82.42
BERTIN	96.47	88.65	80.50
RoBERTa BNE base	96.82	89.90	81.12
RoBERTa BNE large	97.00	90.00	51.62
Task-specific Knowledge Distillation			
ALBETO tiny	96.40	85.05	75.99
ALBETO base-2	96.20	76.75	73.65
ALBETO base-4	96.35	86.40	78.68
ALBETO base-6	96.40	88.45	81.66
ALBETO base-8	96.70	89.75	82.55
ALBETO base-10	96.88	89.95	82.26

Table: Models evaluated on sentence or two sentences classification tasks, results are measured using accuracy on the test set of each dataset.

Task Performance - Sequence Tagging

Model	POS	NER
Fine-tuning		
BETO uncased	97.81	80.85
BETO cased	98.95	87.14
DistilBETO	97.67	78.13
ALBETO tiny	97.34	75.42
ALBETO base	98.21	82.89
ALBETO large	97.98	82.36
ALBETO xlarge	98.43	83.06
ALBETO xxlarge	98.43	83.06
BERTIN	99.02	85.66
RoBERTa BNE base	99.00	86.80
RoBERTa BNE large	61.83	21.47
Task-specific Knowledge Distillation		
ALBETO tiny	97.36	72.51
ALBETO base-2	97.17	69.69
ALBETO base-4	97.60	74.58
ALBETO base-6	97.82	78.41
ALBETO base-8	97.96	80.23
ALBETO base-10	98.00	81.10

Table: Models evaluated on sequence tagging tasks, results are measured using the F1 Score on the test set of each dataset.

Task Performance - Question Answering

Model	MLQA	SQAC	TAR, XQuAD
Fine-tuning			
BETO uncased	64.12 / 40.83	72.22 / 53.45	74.81 / 54.62
BETO cased	67.65 / 43.38	78.65 / 60.94	77.81 / 56.97
DistilBETO	57.97 / 35.50	64.41 / 45.34	66.97 / 46.55
ALBETO tiny	51.84 / 28.28	59.28 / 39.16	66.43 / 45.71
ALBETO base	66.12 / 41.10	77.71 / 59.84	77.18 / 57.05
ALBETO large	65.56 / 40.98	76.36 / 56.54	76.72 / 56.21
ALBETO xlarge	68.26 / 43.76	78.64 / 59.26	80.15 / 59.66
ALBETO xxlarge	70.17 / 45.99	81.49 / 62.67	79.13 / 58.40
BERTIN	66.06 / 42.16	78.42 / 60.05	77.05 / 57.14
RoBERTa BNE base	67.31 / 44.50	80.53 / 62.72	77.16 / 55.46
RoBERTa BNE large	67.69 / 44.88	80.41 / 62.14	77.34 / 56.97
Task-specific Knowledge Distillation			
ALBETO tiny	54.17 / 32.22	63.03 / 43.35	67.47 / 46.13
ALBETO base-2	48.62 / 26.17	58.40 / 39.00	63.41 / 42.35
ALBETO base-4	62.19 / 38.28	71.41 / 52.87	73.31 / 52.43
ALBETO base-6	66.35 / 42.01	76.99 / 59.00	75.59 / 56.72
ALBETO base-8	67.39 / 42.94	77.79 / 59.63	77.89 / 56.72
ALBETO base-10	68.29 / 44.29	79.89 / 62.04	78.21 / 56.21

Table: Models evaluated on question answering datasets, results are noted as F1 Score / Exact Match on the test set of each dataset.

Model Efficiency and Inference Speed



Inference Speed on Common Hardware

Model	Inferences per second	
	CPU	GPU
Fine-tuning		
BETO <i>uncased</i>	3.96	107.19
BETO <i>cased</i>	4.26	109.02
DistilBETO	9.12	217.40
ALBETO <i>tiny</i>	32.53	539.61
ALBETO <i>base</i>	4.50	108.62
ALBETO <i>large</i>	1.29	33.62
ALBETO <i>xlarge</i>	0.35	11.72
ALBETO <i>xxlarge</i>	0.14	6.60
BERTIN	3.99	109.39
RoBERTa BNE <i>base</i>	3.82	107.77
RoBERTa BNE <i>large</i>	1.18	33.65
Task-specific Knowledge Distillation		
ALBETO <i>tiny</i>	32.53	539.61
ALBETO <i>base-2</i>	31.08	625.30
ALBETO <i>base-4</i>	15.16	319.32
ALBETO <i>base-6</i>	10.45	213.53
ALBETO <i>base-8</i>	6.82	160.66
ALBETO <i>base-10</i>	6.01	128.38

Table: The number of inferences per second of models on two different hardware settings, CPU and GPU.

Results - Summary

Model	Parameters	Speedup	Score
Fine-tuning			
BETO <i>uncased</i>	110M	1.00x	81.02
BETO <i>cased</i>	110M	1.00x	84.82
DistilBETO	67M	2.00x	76.73
ALBETO <i>tiny</i>	5M	18.05x	74.97
ALBETO <i>base</i>	12M	0.99x	83.25
ALBETO <i>large</i>	18M	0.28x	82.02
ALBETO <i>xlarge</i>	59M	0.07x	84.13
ALBETO <i>xxlarge</i>	223M	0.03x	85.17
BERTIN	125M	1.00x	83.97
RoBERTa BNE <i>base</i>	125M	1.00x	84.83
RoBERTa BNE <i>large</i>	355M	0.28x	68.42
Task-specific Knowledge Distillation			
ALBETO <i>tiny</i>	5M	18.05x	76.49
ALBETO <i>base-2</i>	12M	5.96x	72.98
ALBETO <i>base-4</i>	12M	2.99x	80.06
ALBETO <i>base-6</i>	12M	1.99x	82.70
ALBETO <i>base-8</i>	12M	1.49x	83.78
ALBETO <i>base-10</i>	12M	1.19x	84.32

Table: The summary of results of every evaluated model in terms of parameters, inference speedup and overall score across tasks. The speedup is relative to BETO models. The score column shows the average of the metrics on all tasks.

Conclusions

Summary of Contributions

We introduced ALBETO and Speedy Gonzales, which are **two novel resources** for the **Spanish NLP** community that were created to **improve** two key aspects of machine learning models, namely **model size** and **inference speed**.

Summary of Contributions

ALBETO:

- Language models that were pre-trained exclusively for the Spanish language, with five different sizes: *tiny*, *base*, *large*, *xlarge*, and *xxlarge*.
- Successfully utilize the weight-shared strategy to achieve greater efficiency in terms of model parameters.
- The *base* model, which is an *uncased* model, outperforms the *uncased* version of BETO while having significantly less parameters and is marginally inferior to other *base*-sized models with a *cased* vocabulary.
- The *xxlarge* model outperforms all other models.

Summary of Contributions

Speedy Gonzales:

- Collection of fast task-specific models trained using Task-specific KD.
- Task-Specific KD is effective in transferring knowledge from a larger model to a lighter and faster model.
- Speedy Gonzales models achieve comparable task performance to most base-sized models while exhibiting enhanced inference speed.
- There exists a trade-off between inference-efficiency of the model and task performance, as observed in the evaluation of the Speedy Gonzales models derived from the ALBETO *base* model.
- Some tasks benefit from the use of larger and more computationally complex models (e.g. QA), while other tasks can be effectively handled by lighter and faster models (e.g. POS, MLDoc).

Limitations and Future Research Directions

- We only evaluated our models on a limited set of tasks.
- Our KD method can be further improved to produce more efficient task-specific language models:
 - Explore alternative KD approaches, such as distilling intermediate layers of the teacher model, in addition to its output.
 - A multi-teacher approach could be studied, in which the models learn from a collection of teacher models rather than just one.
 - Combine with other compression techniques, such as parameter-pruning or quantization.
- There exists a trade-off between model size, inference speed, and task performance, making it challenging to choose an appropriate model without context. It is important to develop metrics to formally assess this balance.

Outcomes

Two publications:

1. ALBETO and DistilBETO: Lightweight Spanish Language Models
 - Cañete et al. [5]
 - Proceedings of the 13th Edition of The Language Resources and Evaluation Conference (LREC), Marseille, France.
 - [Paper](#), [Code](#)
2. Speedy Gonzales: A Collection of Fast Task-Specific Models for Spanish
 - Cañete and Bravo-Marquez [9]
 - Under review.
 - [Code](#)

Models:

- Over 140 models (between pre-trained, fine-tuned and distilled models) publicly available to the research community.
- [Models at the HuggingFace Hub](#)

Light and Fast Language Models for Spanish Through Compression Techniques

José Cañete López

Supervisor: Felipe Bravo-Marquez
Department of Computer Science
Universidad de Chile

April 4, 2023

Machine Learning

- Machine Learning is a subfield of Computer Science that studies the question on how to build algorithms that can automatically improve through experience [23].
- Two paradigms: unsupervised and supervised.
- Unsupervised: According to Jordan and Mitchell [23], is the "*analysis of unlabeled data under assumptions about structural properties of the data (e.g., algebraic, combinatorial, or probabilistic)*". A common example is Clustering.
- Supervised: We use a set of data samples with the form of (x, y) , where x is called an example and y is called its label. The goal is to learn a parameterized function $f(x)$ that maps from x to y and that generalizes to unseen pairs (x^*, y^*) .

Representation Learning

- In classical Machine Learning, the input examples x were represented as feature vectors, which were manually engineered, in a process called "feature engineering", by domain experts who possessed knowledge on the specific task at hand.
- More recently, not only a function $f(x)$ is learned but also a rich and useful representation x is learned from a simpler representation of the data.

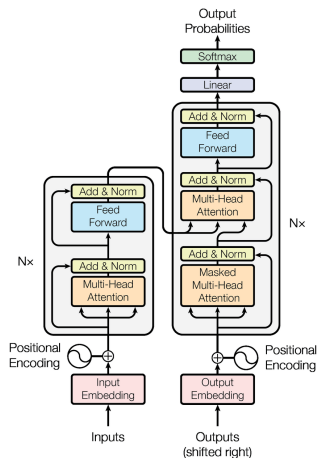
Transfer Learning

- Key idea: reutilize the knowledge (or the representation) learned in one very general task, to another more specific task.
- In Computer Vision (CV), a model is initially trained on a vast labeled dataset with distinct categories known as ImageNet [16, 40, 39]. The model is then fine-tuned or re-trained to perform other tasks or classify objects in categories not present in ImageNet.
- In Natural Language Processing (NLP), where a model is pre-trained for tasks like Language Modeling [36, 37, 4] or Masked Language Modeling [17, 24, 28]. Subsequently, the pre-trained model is fine-tuned for several other tasks like sentiment analysis, question answering, and document classification.

Representations of Text

- Word Embeddings are a mathematical mapping from a word (a discrete symbol) to a continuous vector of dimensionality d .
- First, sparse vectors. More recently, learned dense vectors. (e.g. Word2Vec [30], GloVe [33], FastText [3]).
- One major limitation: Polysemy. They are fixed vectors, meaning that a word is represented identically, regardless of its context.
- To overcome this limitation, contextual word representations are used nowadays. These representations not only used a fixed embedding layer, but also deep neural networks, to account for the complete context of a text in the calculation of a representation of a word.
- The first contextual representations used RNNs [35] as neural network architecture and then were replaced in favor of Transformers [17].

Transformer



Scaled Dot-Product Attention

$W^Q \in \mathbb{R}^{d_{model} \times d_q}$, $W^K \in \mathbb{R}^{d_{model} \times d_k}$, $W^V \in \mathbb{R}^{d_{model} \times d_v}$.
 $Q = XW^Q$, $K = XW^K$, and $V = XW^V$.

$$Attention(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

$$MultiHeadAttention = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O$$

Where, $\text{head}_i = \text{Attention}(XW_i^Q, XW_i^K, XW_i^V)$ and $W^O \in \mathbb{R}^{hd_v \times d_{model}}$.

Figure: The Transformer architecture by Vaswani et al. [48].

Evaluation Tasks - Document Classification - MLDoc

- Assigning a document to a specific category based on its underlying semantic meaning.
- The primary objective of Document Classification is to facilitate efficient information retrieval and management.
- Spanish subset of MLDoc [42]
 - A comprehensive multilingual dataset comprising documents in eight languages.
 - It is derived from the widely used Reuters Corpus [26].
 - Four distinct categories: Corporate/Industrial, Economics, Government/Social, and Markets.

Evaluation Tasks - Document Classification - MLDoc - Example

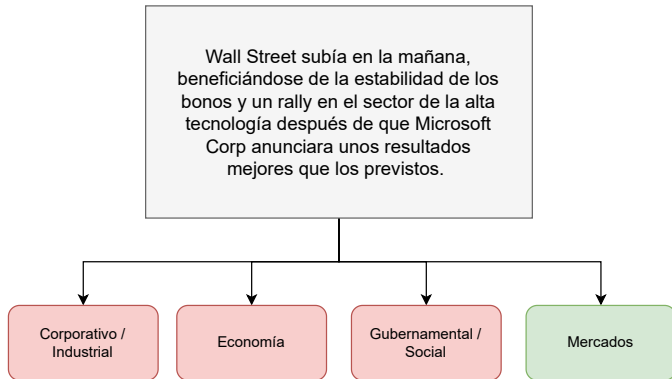


Figure: An example of document classification. Taken from the MLDoc [42] dataset.

Evaluation Tasks - Paraphrase Identification - PAWS-X

- Determine whether two given sentences possess the same underlying semantic meaning.
- Spanish subset of PAWS-X [52]
 - It is a translation of the PAWS [53] dataset in six different languages.
 - The training set of PAWS-X has been machine translated, while the validation and test sets were professionally translated by human experts.

Evaluation Tasks - Paraphrase Identification - PAWS-X - Example

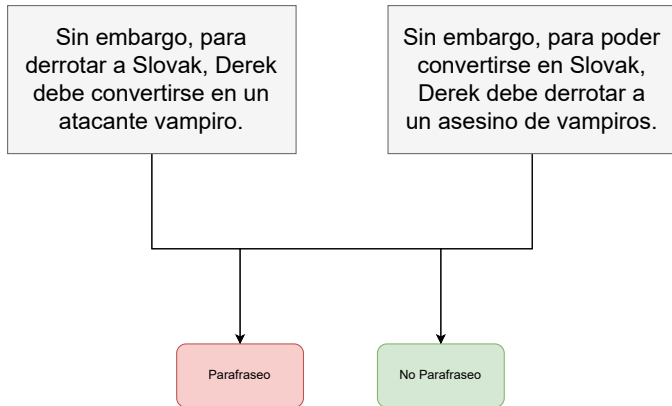


Figure: An example of the paraphrase identification task. Taken from the PAWS-X [52] dataset.

Evaluation Tasks - Natural Language Inference - XNLI

- Determining the logical relationship between two given sentences, namely a "premise" and an "hypothesis". Specifically, the task requires inferring whether the premise entails, contradicts, or is neutral to the hypothesis.
- Spanish subset of XNLI [11]
 - It is a translation of the MultiNLI [51] to 15 different languages.
 - Offers a machine-translated training set while the validation and test sets have been professionally translated.

Evaluation Tasks - Natural Language Inference - XNLI - Example

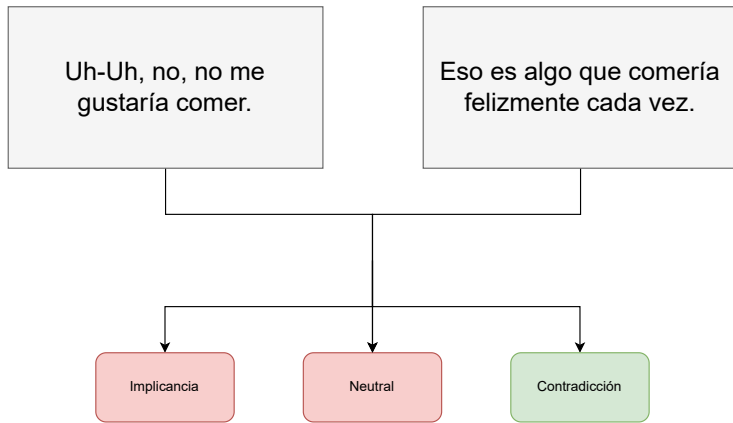


Figure: An example of natural language inference. Taken from the XNLI [11] dataset.

Evaluation Tasks - Part-of-Speech Tagging - POS

- Task that aims to assign each word in a sentence its corresponding syntactic category.
- The syntactic categories are based on the grammatical function of the word and include, among others, nouns, verbs, adjectives, adverbs, and pronouns.
- The dataset used was AnCora [45] which is included on the Spanish part of Universal Dependencies [13] Treebank.

Evaluation Tasks - Part-of-Speech Tagging - POS - Example

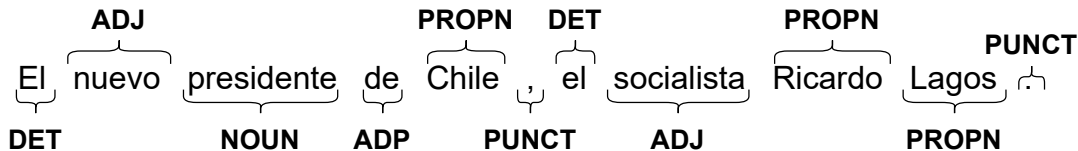



Figure: An example of the Part-of-Speech tagging task. Taken from the AnCora [45] dataset.

Evaluation Tasks - Named Entity Recognition - NER

- Involves identifying and classifying named entities within a text according to their corresponding types.
- It is essential in NLP as it enables computers to extract relevant information from unstructured text data, which can be used for a range of downstream applications.
- Named entities are typically classified into categories such as people, places, organizations, or miscellaneous entities.
- Entities may consist of multiple words. This complexity requires the adoption of the BIO annotation scheme in NER datasets, where each word is labeled as either the beginning (B) of an entity, inside (I) an entity, or outside (O) of any entity.
- Spanish subset of the CoNLL-2002 shared task dataset [46].

Evaluation Tasks - Named Entity Recognition - NER - Example

En su lugar entró el chileno Iván Luis Zamorano



PER

Figure: An example of named entity recognition. Taken from the CoNLL2002 NER [46] dataset.

Evaluation Tasks - Question Answering - MLQA - SQAC - TAR/XQuAD

- Extractive Question Answering: which aims to extract a span of words from a given context text that fully answers a question posed about that context.
- Spanish subset of MLQA [27]
 - Multilingual dataset, created by translating English QA instances into 6 languages.
 - The dataset provides a validation and a test set for each language, as well as a machine-translated version of the SQuAD v1.1 [38] as a training set.
- TAR [6] + XQuAD [2]
 - TAR [6] is another machine-translated dataset from SQuAD v1.1 to Spanish.
 - XQuAD [2] provides a test set that was obtained from SQuAD v1.1 and professionally translated into 11 different languages, including Spanish.
 - Following the setup proposed by [10], we combined the train and validation sets from TAR and the Spanish test set from XQuAD as a single evaluation dataset.
- SQAC [20]
 - May offer a more valuable resource for addressing Spanish language-related challenges, since it is the only one specifically designed for the Spanish language.

Evaluation Tasks - Question Answering - MLQA - SQAC - TAR/XQuAD - Example

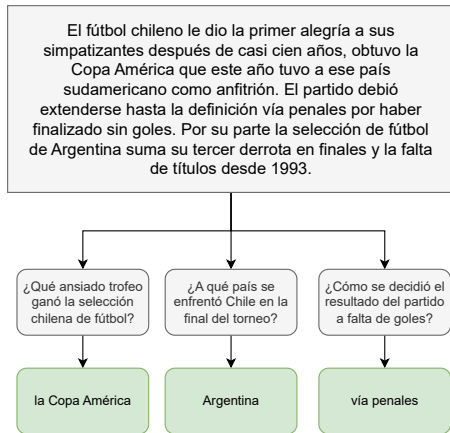


Figure: An example of question answering. Taken from the SQAC [20] dataset.

Evaluation Metrics - Accuracy

Accuracy is a metric that calculates the ratio of correct predictions to the total number of predictions made by a model. It can be expressed mathematically as:

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{All Predictions}}$$

Evaluation Metrics - F1 Score

In the context of binary classification, *Precision* is defined as the proportion of examples classified as positive that are truly positive. This can be expressed as:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Recall is defined as the proportion of truly positive examples that are correctly classified. This can be expressed as:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

The *F1 Score* is then defined as the harmonic mean of *Precision* and *Recall*, given by:

$$\text{F1 Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Evaluation Metrics - Exact Match

In the case of Question Answering, the Exact Match metric compares the predicted answer string, p_s , with the correct answer string, c_s . The Exact Match for a single example is defined as:

$$\text{Exact Match}_{\text{single}} = \begin{cases} 1, & \text{if } p_s == c_s \\ 0, & \text{otherwise} \end{cases}$$

The Exact Match for a collection of pairs $(p_s, c_s) \in A$ is then defined as the average of the Exact Match for a single example, expressed as:

$$\text{Exact Match} = \sum_{(p_s, c_s) \in A} \frac{\text{Exact Match}_{\text{single}}(p_s, c_s)}{|A|}$$

ALBETO - Dataset - SUC

- General domain corpora.
- Same corpus [8] used on BETO [10].
- 300M lines, 3B tokens, 18.4B chars.
- Sources: Spanish Wikis (dump of April 2019), Books, News, Subtitles, European Parliament, TED Talks, etc.

ALBETO - Preprocessing

- Identical to BETO [10] and very simple.
- Removing URLs and listings.
- Removing multiple whitespaces.
- Lowercase.

ALBETO - Pre-training Details

- MLM and SOP.
- Single TPU v3-8 for each model.
- A maximum sequence length of 512 was used for pre-training, and the largest multiple of 64 that fit in the TPU memory was selected as the batch size.
- We experienced divergence in the loss on the *large* and *xlarge* models, this issue forced to stop the training and restart it from an earlier checkpoint with a slightly lower learning rate.

Model	Learning Rate	Batch Size	Warmup Ratio	Warmup Steps	Total Steps	Training Time (days)
ALBETO <i>tiny</i>	1.25e-3	2,048	1.25e-2	125,000	8,300,000	58.2
ALBETO <i>base</i>	8.83e-4	960	6.25e-3	53,333	3,650,000	70.4
ALBETO <i>large</i>	6.25e-4	512	3.12e-3	12,500	1,450,000	42.0
ALBETO <i>xlarge</i>	3.12e-4	128	7.81e-4	6,250	2,775,000	64.2
ALBETO <i>xxlarge</i>	3.12e-4	128	7.81e-4	3,125	1,650,000	70.7

Table: Training details of all ALBETO models, which were trained using a single TPU v3-8 each one.

ALBETO - Training Loss - tiny

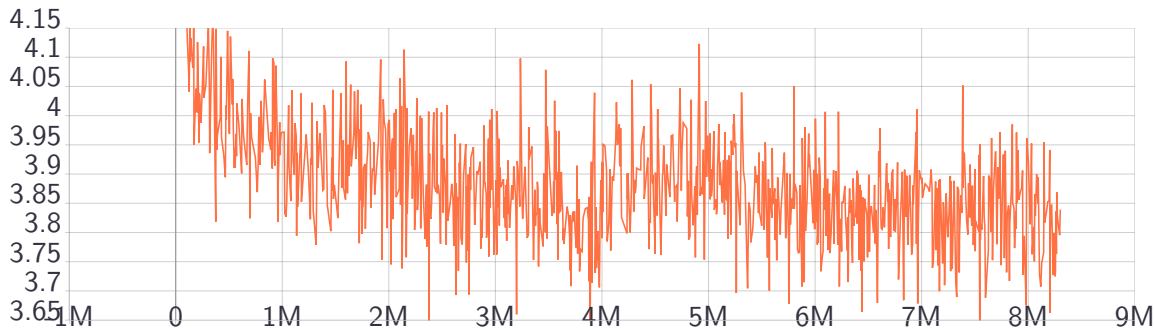


Figure: The progression of the training loss on the ALBETO *tiny* model.

ALBETO - Training Loss - base

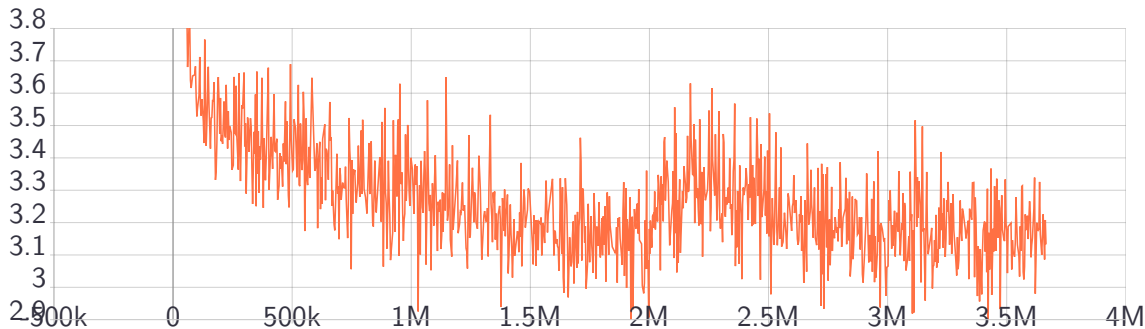


Figure: The progression of the training loss on the ALBETO *base* model.

ALBETO - Training Loss - large

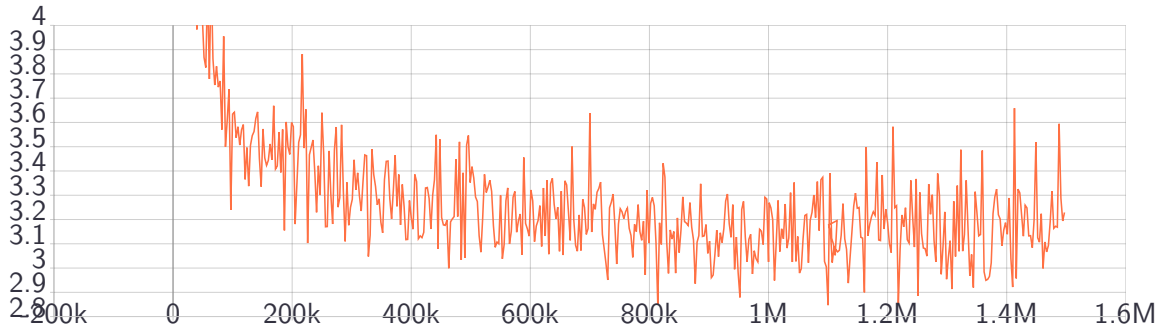


Figure: The progression of the training loss on the ALBETO *large* model.

ALBETO - Training Loss - xlarge

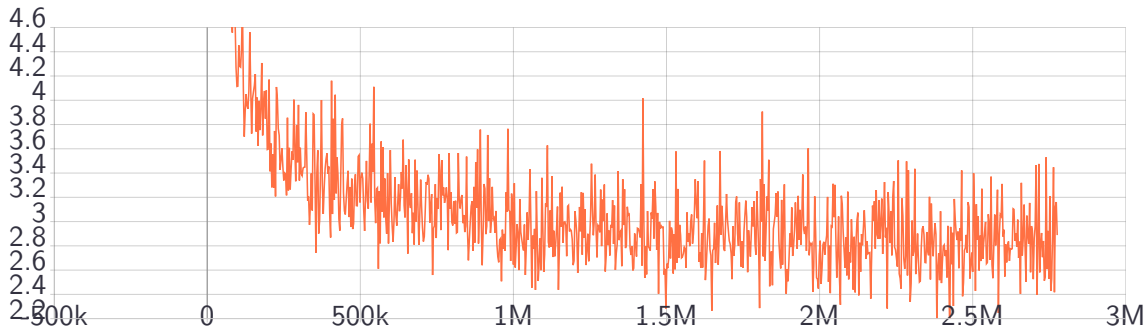


Figure: The progression of the training loss on the ALBETO *xlarge* model.

ALBETO - Training Loss - xxlarge

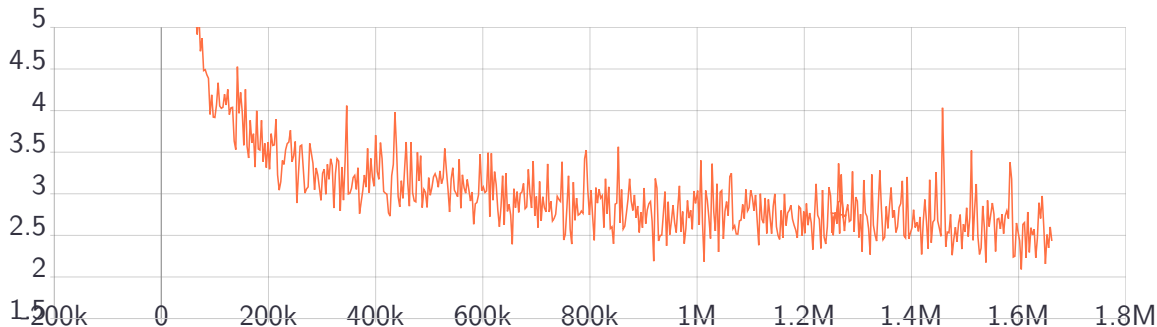


Figure: The progression of the training loss on the ALBETO *xxlarge* model.

ALBETO - Fine-tuning Details

- We conducted a hyperparameter search on BETO, DistilBETO, RoBERTa-BNE, BERTIN, ALBETO *tiny* and *base*, exploring combinations of batch size $\{16, 32, 64\}$, learning rate $\{1e-5, 2e-5, 3e-5, 5e-5\}$, and number of epochs 2, 3, 4.
- For the larger ALBETO models (*large*, *xlarge*, and *xxlarge*), we reduced the learning rates to $\{1e-6, 2e-6, 3e-6, 5e-6\}$ to mitigate numerical instability issues during training.
- These fine-tuning procedures were performed on one to two NVIDIA RTX 3090 GPUs, depending on the model and task.
- To fine-tune the largest models on QA, we utilized two NVIDIA A100 GPUs from the Patagón supercomputer [31].
- We used gradient accumulation in situations where the GPU memory was insufficient to reach the target batch size.

Speedy Gonzales: KD Implementation - Text Classification - Single Sentence

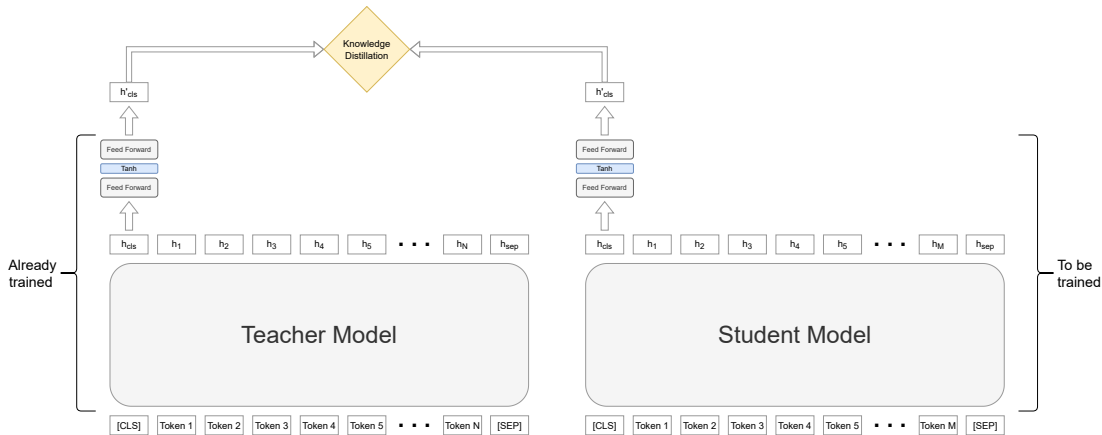


Figure: Implementation of KD for text classifications tasks that use a single sentence as input.

Speedy Gonzales: KD Implementation - Text Classification - Two Sentences

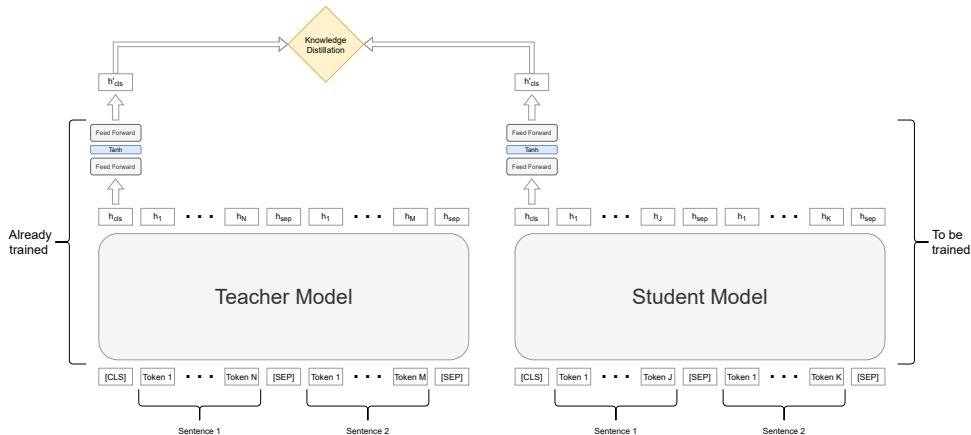


Figure: Implementation of KD for text classifications tasks that use two sentences as input.

Speedy Gonzales: KD Implementation - Sequence Tagging

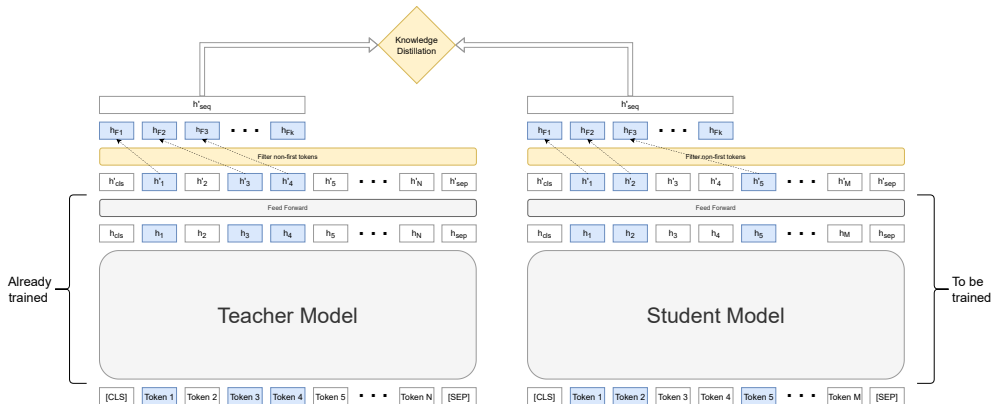


Figure: Implementation of KD for sequence tagging tasks. The tokens marked with the blue color represents the property of being the first token of a word.

Speedy Gonzales: KD Implementation - Sequence Tagging - Same Vocabulary

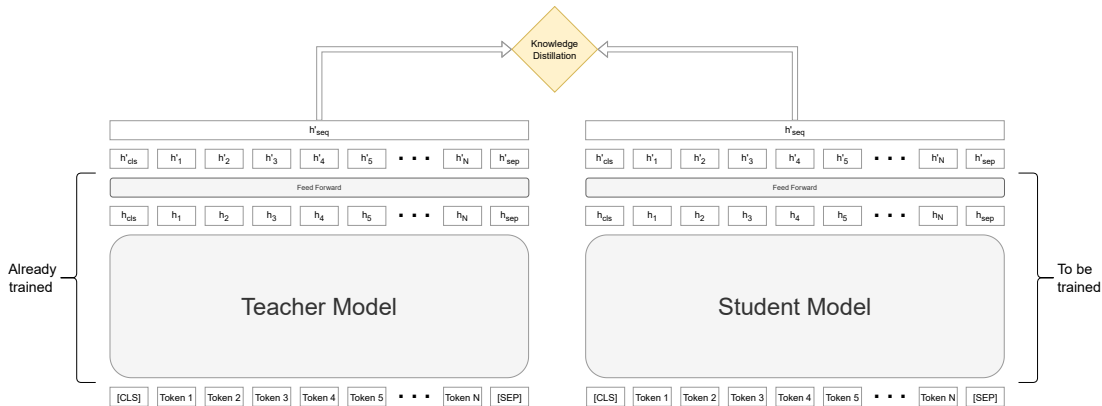


Figure: Implementation of KD for sequence tagging tasks with models that share the same vocabulary.

Speedy Gonzales: KD Implementation - Question Answering

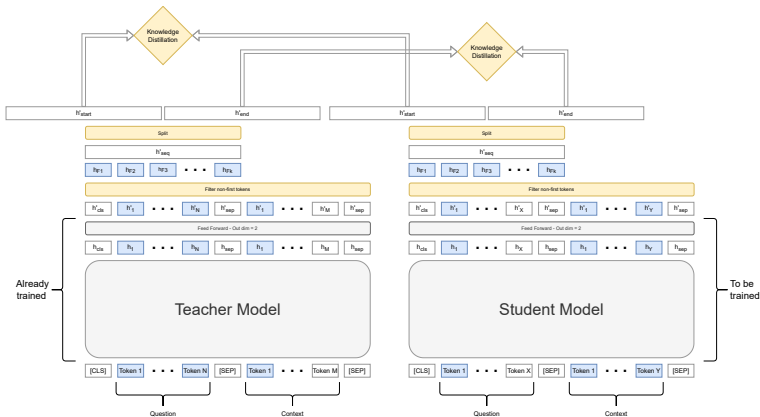


Figure: Implementation of KD for question answering datasets. The tokens marked with the blue color represents the property of being the first token of a word.

Speedy Gonzales: KD Implementation - Question Answering - Same Vocabulary

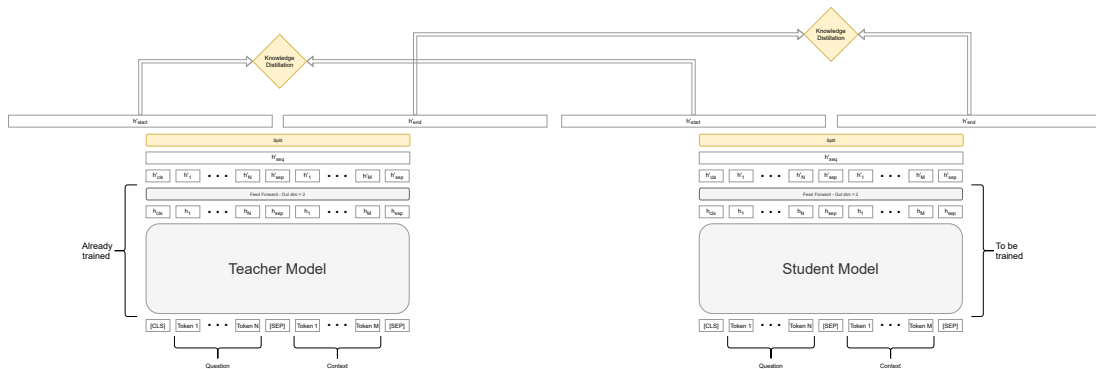


Figure: Implementation of KD for question answering datasets with models that share the same vocabulary.

Speedy Gonzales: Other details

- Our code uses Python and PyTorch [32].
- To measure MACs we used the THOP¹ library.
- We conducted initial experiments utilizing three distinct loss functions: mean-squared error loss, cross-entropy loss, and KL-divergence loss. We varied the parameters α and T across these losses using Optuna [1]. The outcomes of these experiments revealed that the optimal settings were $\alpha = 0$ and $T = 1$. Although all three losses yielded satisfactory outcomes with this configuration, KL-divergence produced marginally superior results.

¹<https://github.com/Lyken17/pytorch-OpCounter>

Speedy Gonzales: Selected Teacher Models

Dataset	Teacher Model
MLDoc	RoBERTa BNE <i>large</i>
PAWS-X	ALBETO <i>xxlarge</i>
XNLI	ALBETO <i>xxlarge</i>
POS	RoBERTa BNE <i>base</i>
NER	RoBERTa BNE <i>base</i>
MLQA	ALBETO <i>xxlarge</i>
SQAC	ALBETO <i>xxlarge</i>
TAR / XQuAD	ALBETO <i>xxlarge</i>

Table: The teacher models selected for each task.

Table 10 presents the teacher models selected for each task. The selection process is based on the lowest validation loss achieved among the candidate teacher models that were fine-tuned for each task.

Bibliography I

- [1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis, editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 2623–2631. ACM, 2019. doi: 10.1145/3292500.3330701. URL <https://doi.org/10.1145/3292500.3330701>.
- [2] Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.421. URL <https://aclanthology.org/2020.acl-main.421>.

Bibliography II

- [3] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.
- [4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

Bibliography III

URL <https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf>.

- [5] José Cañete, Sebastian Donoso, Felipe Bravo-Marquez, Andrés Carvallo, and Vladimir Araujo. ALBETO and DistilBETO: Lightweight Spanish language models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4291–4298, Marseille, France, June 2022. European Language Resources Association. URL <https://aclanthology.org/2022.lrec-1.457>.
- [6] Casimiro Pio Carrino, Marta R. Costa-jussà, and José A. R. Fonollosa. Automatic Spanish translation of SQuAD dataset for multi-lingual question answering. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5515–5523, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL <https://aclanthology.org/2020.lrec-1.677>.

Bibliography IV

- [7] Casimiro Pio Carrino, Joan Llop, Marc Pàmies, Asier Gutiérrez-Fandiño, Jordi Armengol-Estapé, Joaquín Silveira-Ocampo, Alfonso Valencia, Aitor Gonzalez-Agirre, and Marta Villegas. Pretrained biomedical language models for clinical NLP in Spanish. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 193–199, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.bionlp-1.19. URL <https://aclanthology.org/2022.bionlp-1.19>.
- [8] José Cañete. Compilation of Large Spanish Unannotated Corpora, May 2019. URL <https://doi.org/10.5281/zenodo.3247731>.
- [9] José Cañete and Felipe Bravo-Marquez. Speedy gonzales: A collection of fast task-specific models for spanish, 2023.

Bibliography V

- [10] José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. Spanish pre-trained bert model and evaluation data. In *PML4DC at ICLR 2020*, 2020.
- [11] Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1269. URL <https://aclanthology.org/D18-1269>.

Bibliography VI

- [12] Javier de la Rosa, Eduardo G Ponferrada, Paulo Villegas, Pablo González de Prado Salas, Manu Romero, and Maria Grandury. Bertin: Efficient pre-training of a spanish language model using perplexity sampling. *Procesamiento del Lenguaje Natural*, 68(0):13–23, 2022. ISSN 1989-7553. URL <http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/6403>.
- [13] Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. Universal Dependencies. *Computational Linguistics*, 47(2):255–308, June 2021. doi: 10.1162/coli_a_00402. URL <https://aclanthology.org/2021.cl-2.11>.
- [14] Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. BERTje: A Dutch BERT Model. arXiv:1912.09582, December 2019. URL <http://arxiv.org/abs/1912.09582>.

Bibliography VII

- [15] Pieter Delobelle, Thomas Winters, and Bettina Berendt. RobBERT: a Dutch RoBERTa-based Language Model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3255–3265, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.292. URL <https://aclanthology.org/2020.findings-emnlp.292>.
- [16] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Bibliography VIII

- [17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL <https://aclanthology.org/N19-1423>.
- [18] Sebastián Alejandro Donoso. Entrenamiento y evaluación de modelos pequeños de lenguaje natural basado en métodos de autoatención. 2021. URL <https://repositorio.uchile.cl/handle/2250/183444>.
- [19] Asier Gutiérrez-Fandiño, Jordi Armengol-Estapé, Aitor Gonzalez-Agirre, and Marta Villegas. Spanish legalese language model and corpora, 2021.

Bibliography IX

- [20] Asier Gutiérrez-Fandiño, Jordi Armengol Estapé, Marc Pàmies, Joan Llop Palao, Joaquin Silveira Ocampo, Casimiro Pio Carrino, Carme Armentano Oller, Carlos Rodriguez Penagos, Aitor Gonzalez Agirre, and Marta Villegas. Maria: Spanish language models. *Procesamiento del Lenguaje Natural*, 68, 2022. ISSN 1135-5948. doi: 10.26342/2022-68-3. URL <https://upcommons.upc.edu/handle/2117/367156#.YyMTB4X9A-0.mendeley>.
- [21] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

Bibliography X

- [22] Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.372. URL <https://aclanthology.org/2020.findings-emnlp.372>.
- [23] Michael I Jordan and Tom M Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.
- [24] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL <https://openreview.net/forum?id=H1eA7AEtvS>.

Bibliography XI

- [25] Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. FlauBERT: Unsupervised language model pre-training for French. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL <https://aclanthology.org/2020.lrec-1.302>.
- [26] David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. RCV1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5:361–397, 2004. URL <http://jmlr.org/papers/volume5/lewis04a/lewis04a.pdf>.

Bibliography XII

- [27] Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.653. URL <https://aclanthology.org/2020.acl-main.653>.
- [28] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. URL <http://arxiv.org/abs/1907.11692>.

Bibliography XIII

- [29] Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.645. URL <https://aclanthology.org/2020.acl-main.645>.
- [30] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.
- [31] Austral University of Chile. Patagón supercomputer, 2021. URL <https://patagon.uach.cl>.

Bibliography XIV

- [32] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL <https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf>.
- [33] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. URL <http://www.aclweb.org/anthology/D14-1162>.

Bibliography XV

- [34] Juan Manuel Pérez, Damián Ariel Furman, Laura Alonso Alemany, and Franco M. Luque. RoBERTuito: a pre-trained language model for social media text in Spanish. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 7235–7243, Marseille, France, June 2022. European Language Resources Association. URL <https://aclanthology.org/2022.lrec-1.785>.
- [35] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL <https://aclanthology.org/N18-1202>.

Bibliography XVI

- [36] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.
- [37] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.
- [38] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. URL <https://aclanthology.org/D16-1264>.
- [39] Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses, 2021.

Bibliography XVII

- [40] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115: 211–252, 2015.
- [41] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019. URL <http://arxiv.org/abs/1910.01108>.
- [42] Holger Schwenk and Xian Li. A corpus for multilingual document classification in eight languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL <https://aclanthology.org/L18-1560>.

Bibliography XVIII

- [43] Alejandro Vaca Serrano, Guillem Garcia Subies, Helena Montoro Zamorano, Nuria Aldama Garcia, Doaa Samy, David Betancur Sanchez, Antonio Moreno Sandoval, Marta Guerrero Nieto, and Alvaro Barbero Jimenez. Rigoberta: A state-of-the-art language model for spanish. *arXiv preprint arXiv:2205.10233*, 2022.
- [44] Raphael Tang, Yao Lu, Linqing Liu, Lili Mou, Olga Vechtomova, and Jimmy Lin. Distilling task-specific knowledge from BERT into simple neural networks. *CoRR*, abs/1903.12136, 2019. URL <http://arxiv.org/abs/1903.12136>.
- [45] Mariona Taulé, M. Antònia Martí, and Marta Recasens. AnCora: Multilevel annotated corpora for Catalan and Spanish. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May 2008. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2008/pdf/35_paper.pdf.

Bibliography XIX

- [46] Erik F. Tjong Kim Sang. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*, 2002. URL <https://aclanthology.org/W02-2024>.
- [47] Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962v2*, 2019.
- [48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [49] Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. Multilingual is not enough: Bert for finnish. *arXiv preprint arXiv:1912.07076*, 2019.

Bibliography XX

- [50] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS'20*, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.
- [51] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL <https://aclanthology.org/N18-1101>.

Bibliography XXI

- [52] Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3687–3692, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1382. URL <https://aclanthology.org/D19-1382>.
- [53] Yuan Zhang, Jason Baldridge, and Luheng He. PAWS: Paraphrase adversaries from word scrambling. 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 1298–1308, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1131. URL <https://aclanthology.org/N19-1131>.

Light and Fast Language Models for Spanish Through Compression Techniques

José Cañete López

Supervisor: Felipe Bravo-Marquez
Department of Computer Science
Universidad de Chile

April 4, 2023