Comparative Guide to BERT, RoBERTa, DistilBERT, and ALBERT

August 1, 2025

Contents

1 Background and Motivation

Transformers revolutionised natural-language processing (NLP) in 2017, but the 2018 release of **BERT** (*Bidirectional Encoder Representations from Transformers*) marked the first time a bidirectionally pre-trained encoder became the default starting point for almost every language-understanding task. Since then, several variants have emerged—each optimising for speed, compute, parameter efficiency, or downstream performance. This document clarifies how four widely-used models differ and when to choose each.

2 High-Level Comparison

Table 1: Key characteristics of the four models (base variants where applicable).

Model	**Parameters**	**Corpus Size**	**Pre-training Objective(s)**	**Notable Design / Training Tweaks**
BERT (2018)	110 M (Base)	16GB BookWiki	Masked-LM (MLM) + Next-Sentence Prediction (NSP)	Vanilla Transformer encoder, WordPiece vocab of 30k.
RoBERTa (2019)	125 M	160GB (CC-News, OpenWebText, Stories, Books)	MLM *only* (dynamic masking)	No NSP; larger batch ($\sim$8k seq), longer training, byte-level BPE.
DistilBERT (2019)	66 M	Inherited from BERT	MLM (student vs. teacher logits)	Knowledge distillation (temperature 2); 40% fewer params, 60% faster.
ALBERT (2020)	12 M (Base)	BookWiki + 158GB raw text	MLM + Sentence-Order Prediction (SOP)	Factorised embedding layer + cross-layer parameter sharing; `proj_dim` $\ll$ hidden.

Interpretation.

- *Size and speed.* DistilBERT and ALBERT aggressively shrink parameters; RoBERTa purposefully increases compute to maximise accuracy.
- *Objective tweaks.* Removing or replacing NSP yielded measurable gains; dynamic masking (RoBERTa) gives better token coverage.

- *Data scale.* More diverse text corpora enable RoBERTa to generalise better out-of-the-box, especially for domain-generic tasks.

3 Model-by-Model Details

3.1 BERT

- **Bidirectionality.** All tokens attend to both left and right context during pre-training.
- **NSP rationale.** Encourages learning inter-sentence coherence, later shown to be replaceable.
- **Hidden size.** 768 (Base), 1024 (Large).
- **When to use.** Strong baseline; well-supported in most libraries. Still preferred for pedagogical demos or if downstream data is *small* and you need stable, replicable baselines.

3.2 RoBERTa

- **No NSP.** Facebook AI showed NSP hurts longer-sequence accuracy; removing it speeds convergence.
- **Dynamic masking.** Each sentence is masked differently every epoch, creating $\approx 10\times$ more innate examples.
- **When to use.**
  (a) You want the *best zero-shot or few-shot performance* among BERT-style encoders without changing architecture.
  (b) You can spare extra compute/RAM at inference (parameters $\uparrow$, max-length $514 \rightarrow 1{,}024$ in many checkpoints).

3.3 DistilBERT

- **Distillation scheme.** Student learns from teacher (BERT) logits + true MLM labels simultaneously.
- **Speedup.** $\sim$60% faster on CPU, 40% smaller memory footprint.
- **Accuracy trade-off.** Only 1–2 pp lower on GLUE while being mobile-friendly.
- **When to use.**
  (a) Edge/real-time inference (chat-bots, browser extensions, on-device text classification).
  (b) Batch inference pipelines where throughput is the bottleneck.

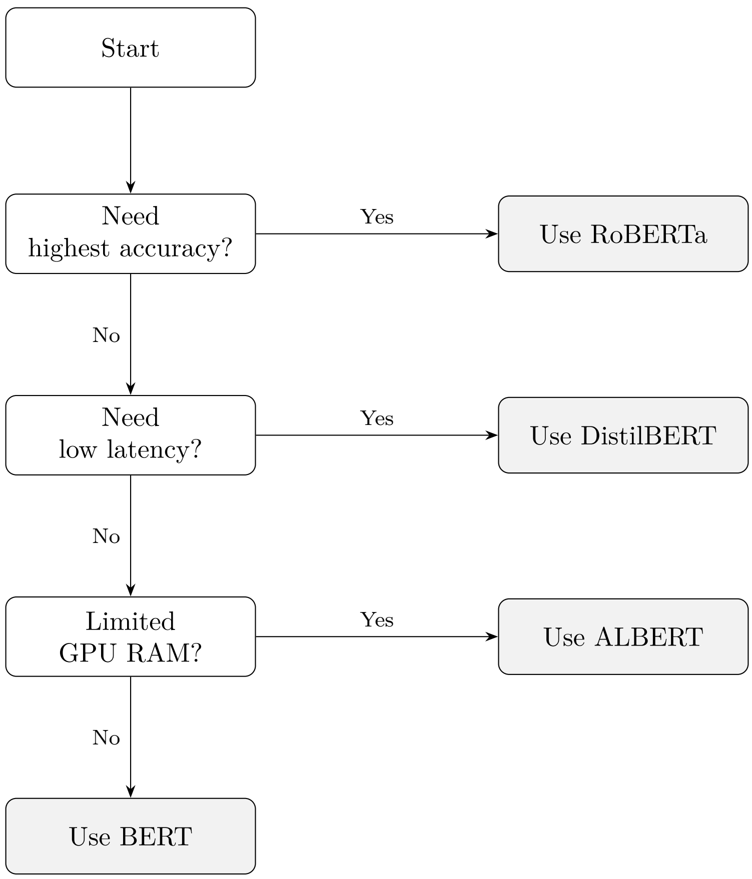
3.4 ALBERT

- **Parameter factorisation.** Decomposes large $V \times H$ embedding into $V \times E$ and $E \times H$ where $E \ll H$.
- **Cross-layer sharing.** All Transformer layers share weights huge memory savings, but same feed-forward compute.
- **Sentence-Order Prediction (SOP).** More robust than NSP for discourse understanding.
- **Scaling law.** Parameters stay low while hidden size grows; ALBERT-xxlarge sets a GLUE record with just 235 M trainable weights.

- **When to use.** Very deep fine-tuning (hundreds of epochs) where GPU memory is limited; multi-sentence reasoning tasks (natural language inference, reading comprehension).

4 Decision Framework: Which Model When?

1. **Accuracy & Data Agnostic? RoBERTa** → highest base accuracy, especially when downstream data is diverse or non-domain-specific.
2. **Latency/Memory Sensitive? DistilBERT** → fastest and smallest without manual pruning/quantisation.
3. **Long Fine-Tuning Horizons? ALBERT** → less overfitting, fits in modest GPU RAM; recommend if your corpus has long documents and coherent sentence-order matters.
4. **Educational Baseline or Stable Reproduction? BERT** → canonical reference; abundant tutorials/checkpoints across languages.



5 Practical Fine-Tuning Tips

- Always **lower the learning rate** by 5–10× when switching from BERT to ALBERT or RoBERTa (they converge faster).

- **Freeze first k layers** of RoBERTa if fine-tuning data $< 10\,\mathrm{k}$ examples—mitigates catastrophic forgetting.
- Consider **mixed-precision (FP16/BF16)** for RoBERTa and ALBERT to offset larger sequence lengths.
- For DistilBERT, layer-wise learning-rate decay is less important; its shallower depth benefits from uniform optimisation.