BERT

SICSS Paris '25

Julien Boelaert

CERAPS, Université de Lille

25/06/2025

Outline

BERT: pre-trained models for supervized learning

1. Motivation, uses
2. The BERT model(s)
3. Fine-tuning tricks

1a/ BERT: motivation

BERT: Bidirectional Encoder Representations from Transformers
Masked language model for NLP tasks

- only the encoder half of the Transformer
- pre-trained on massive corpora for **masked language model**
  Essentially, learn to reconstruct corrupted sentences:

  "The [MASK] is [MASK] the mat"

1a/ BERT: motivation

BERT: Bidirectional Encoder Representations from Transformers
Masked language model for NLP tasks

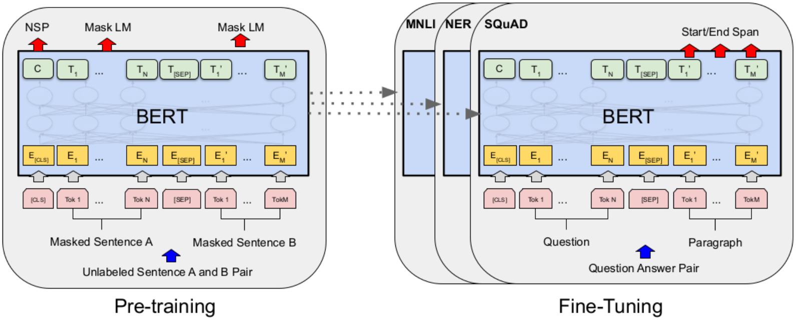
- only the encoder half of the Transformer
- pre-trained on massive corpora for **masked language model**
  Essentially, learn to reconstruct corrupted sentences:

  "The [MASK] is [MASK] the mat"
- fine-tunable for SOTA performance on NLP tasks: sequence classification, token classification, question-answering, language inference, ...
- intuition: the pre-training gives the model a fine "understanding" of semantics, that we can harness for many tasks. A few hundred examples are enough during fine-tuning.

1a/ BERT: motivation

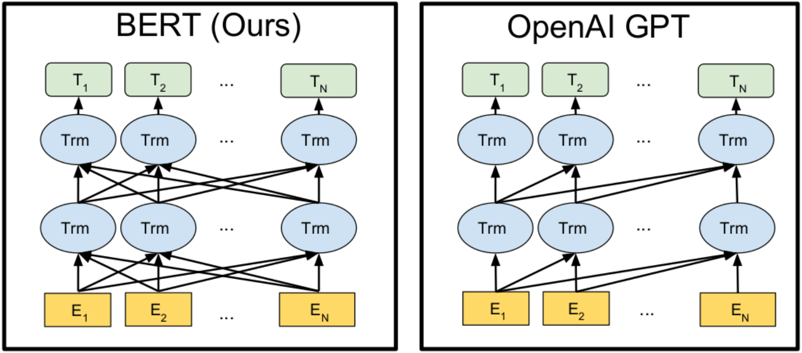
Using BERT:

1/ Download a pre-trained model   2/ Fine-tune it on your data



1a/ BERT: motivation

Why "bidirectional"?



BERT: Each token sees every token

GPT: each token sees only previous tokens

1a/ BERT: motivation

	BERT	**GPT**
Direction	Bidirectional	Left-to-right
Transformer	Encoder	Decoder, generative
Training	Masked language model	Causal language model
Size	Small (100M-500M)	Large (1B-300B)
Speed	Fast (small, one pass)	Slow (large, many tokens)
Energy	Low	High
Use	Fine-tune on task	Prompt, no fine-tuning
Confidential	Yes (local)	Not if API
Reproducible	Yes (model=file) (and deterministic)	Not if API (and stochastic)

Thanks to bidirectionality, BERT is (usually) still SOTA on sequence classification, sequence tagging, etc.

BERT also used for AI-assistant censorship.

1b/ BERT: uses

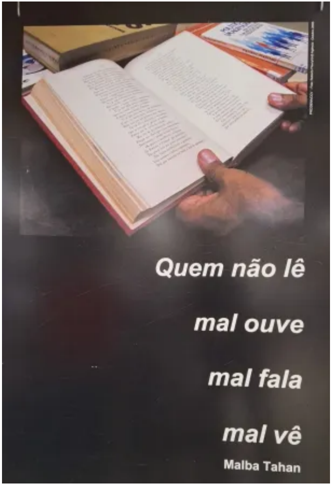
Using BERT in social science:

1. Choose text unit: sentence, paragraph, full?
2. Split data into training and test sets
3. Annotate some training data
4. Fine-tune BERT on annotated texts
5. Repeat 3-4, (*eg* with active learning), until satisfying quality on validation set (= random sample of training set)
6. Annotate test data, evaluate quality (once!)
7. Predict on complete dataset

1b/ BERT: uses

Using BERT in social science:

- Forces you to **read the texts!**
- Annotation process
  → refine your categories
  (active: see ambiguous cases)
- Better performance than generative models on difficult tasks (for now?)



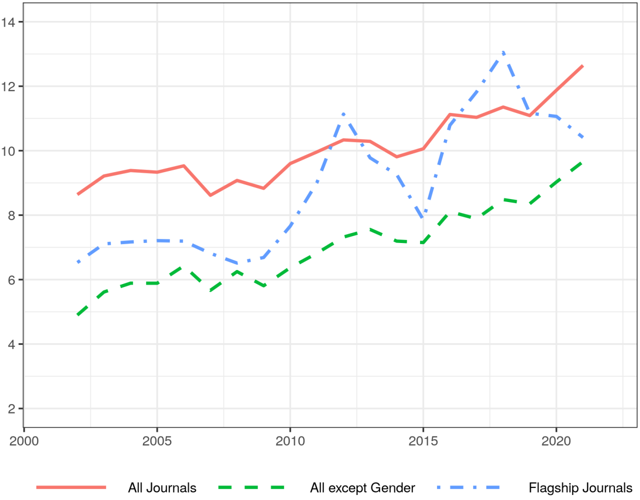
1b/ BERT: uses

Example: Gender in social science
(collective, *Actes de la recherche en sciences sociales*, 2025)

- Dataset: 50,000 articles, from 120 French social science journals, 2001-2022
- Unit: abstract (paragraph)
- Annotate : gender or not (600-2000 tags)
- Fine-tune BERT (F1-score .94 on test set)
- Predict on whole dataset

1b/ BERT: uses

Example: Gender in social science
(collective, *Actes de la recherche en sciences sociales*, 2025)



2/ BERT models

2/ BERT models

Original BERT: Devlin et al. (Google, 2018)

- Pre-trained on two simultaneous tasks:
  - MLM (randomly mask 15% of tokens)
  - next sentence prediction: sentences are fed 2 by 2 (with special separator and context embedding), learn to predict if they follow each other or not (abandoned in later models)

All BERTs:

- Fine-tuned by adding special trainable "head" for each task, on output embeddings (fine-tune whole model):
  - sequence classification: MLP + softmax on the "[CLS]" output
  - token classification: MLP + softmax on each token output

2/ BERT models

Several variants, many improvements in architecture and training:

- **RoBERTa** (Facebook, 2019): First model to surpass Human baselines on GLUE
- **DeBERTa** (Microsoft, 2020): First model to surpass Human baselines on superGLUE (+ v2, v3)
- multilingual BERT, distilBERT, language-specific (*eg* FlauBERT), domain-specific (*eg* EconBERT) ...
- **ModernBERT** (answerdotai, 2025): Faster, cues from generative models, long context (8k vs 512)

2/ BERT models

Model sizes:

- Most models come in several sizes: (small), base, large
- The bigger the model, the better the performance
- Bigger models need more data for good fine-tuning

2/ BERT models

Model sizes:

- Most models come in several sizes: (small), base, large
- The bigger the model, the better the performance
- Bigger models need more data for good fine-tuning

	Model	IR (DPR)			IR (ColBERT)		NLU	Code	
		BEIR	MLDR_{OOD}	MLDR_{ID}	BEIR	MLDR_{OOD}	GLUE	CSN	SQA
Base	BERT	38.9	23.9	32.2	49.0	28.1	84.7	41.2	59.5
	RoBERTa	37.7	22.9	32.8	48.7	28.2	86.4	44.3	59.6
	DeBERTaV3	20.2	5.4	13.4	47.1	21.9	88.1	17.5	18.6
	NomicBERT	41.0	26.7	30.3	49.9	61.3	84.0	41.6	61.4
	GTE-en-MLM	41.4	**34.3**	**44.4**	48.2	69.3	85.6	44.9	71.4
	ModernBERT	**41.6**	27.4	44.0	**51.3**	**80.2**	**88.5**	**56.4**	**73.6**
Large	BERT	38.9	23.3	31.7	49.5	28.5	85.2	41.6	60.8
	RoBERTa	41.4	22.6	36.1	49.8	28.8	88.9	47.3	68.1
	DeBERTaV3	25.6	7.1	19.2	46.7	23.0	**91.4**	21.2	19.7
	GTE-en-MLM	42.5	**36.4**	**48.9**	50.7	71.3	87.6	40.5	66.9
	ModernBERT	**44.0**	34.3	48.6	**52.4**	**80.4**	90.4	**59.5**	**83.9**

2/ BERT models

Another useful variant: **SBERT**, aka sentence-BERT (Reimers and Gurevych 2019)

$\rightarrow$ **Static** sentence embeddings

- **Motivation**:
  - raw static BERT embeddings (w/o fine-tuning) perform worse than word2vec embeddings for downstream tasks!
  - BERT two-sentence similarity is inefficient ($n(n-1)/2$ runs)

2/ BERT models

Another useful variant: **SBERT**, aka sentence-BERT (Reimers and Gurevych 2019)

$\rightarrow$ **Static** sentence embeddings

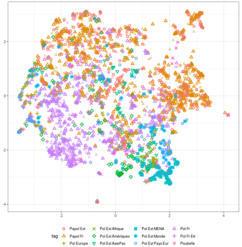
- **Motivation**:
  - raw static BERT embeddings (w/o fine-tuning) perform worse than word2vec embeddings for downstream tasks!
  - BERT two-sentence similarity is inefficient ($n(n-1)/2$ runs)
- **Solution**: fine-tune a BERT model for pairwise similarities (siamese networks, NLI: entailment / contradiction / neutral)
- **Result**: high-quality, fast to compute sentence embeddings
- **Use**: fast classification, visualization, clustering, retrieval

Current SBERT SOTA: AliBaba *gte-multilingual-base* (2025)
$\rightarrow$ 70 languages, 8192-tokens context

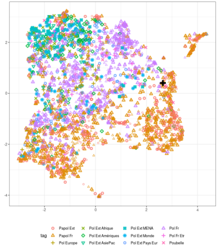
2/ BERT models

Visualization example: UMAP of DTM, fastText and SBERT (French press articles)

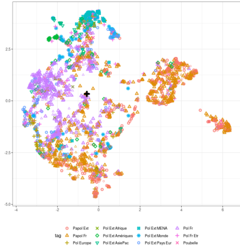
DTM (tf-idf)



fastText



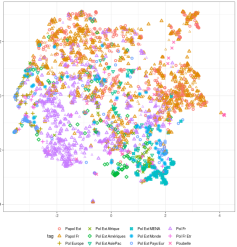
SBERT



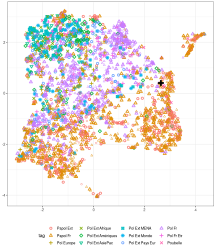
2/ BERT models

Visualization example: UMAP of DTM, fastText and SBERT (French press articles)

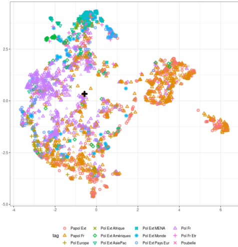
DTM (tf-idf)



fastText



SBERT



NB: Superiority of supervized learning
(fine-tuned BERT classifications $\neq$ static BERT clusters)

2/ BERT models

Building on SBERT: BERTopic for topic modeling (Grootendorst 2022)

Algorithm:

1. SBERT on texts (large embeddings)
2. UMAP on embeddings (100s dims $\rightarrow$ 2D)
3. HDBScan clustering on 2D coordinates
4. Interpretation with topic-wise TF-IDF

Python: bertopic library, modular (possible to replace UMAP and HDBScan by others)

2/ BERT models

Building on SBERT: BERTopic for topic modeling (Grootendorst 2022)

Algorithm:

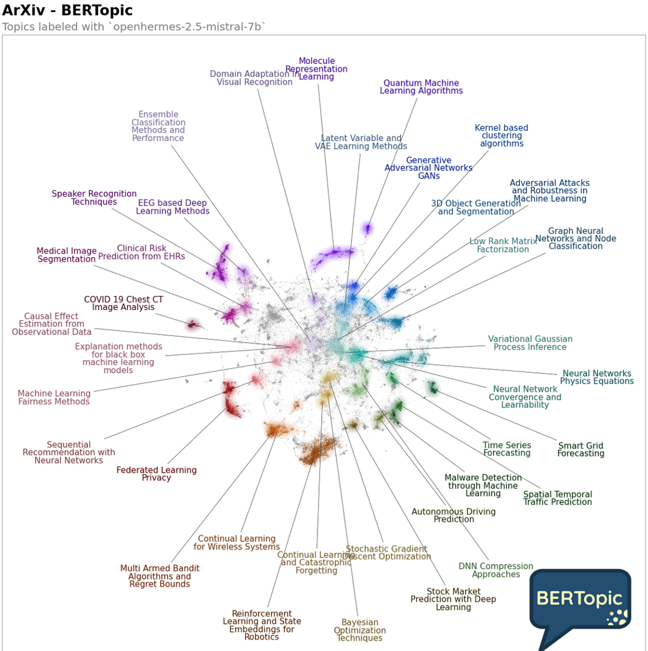
1. SBERT on texts (large embeddings)
2. UMAP on embeddings (100s dims $\rightarrow$ 2D)
3. HDBScan clustering on 2D coordinates
4. Interpretation with topic-wise TF-IDF

Python: bertopic library, modular (possible to replace UMAP and HDBScan by others)

Alternative to LDA topic modeling:

- BERT-level language "understanding" vs bag of words
- Fast, even for large datasets
- No underlying formal statistical model

2/ BERT models



3/ Fine-tuning tricks

3/ Fine-tuning tricks

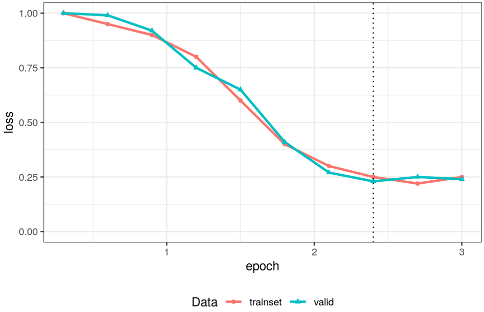
Fine-tuning BERT models can be tricky.

Main hyper-parameters:

	Meaning	Standard
Epochs	How many times to see whole dataset	3
Step size	How much to change at each step	2e-5
Minibatch	How many texts to see at each step	16-32
Weight decay	How much regularization	.01

Key for hyper-parameters choice: examine learning curves (training set vs validation set)

3/ Fine-tuning tricks

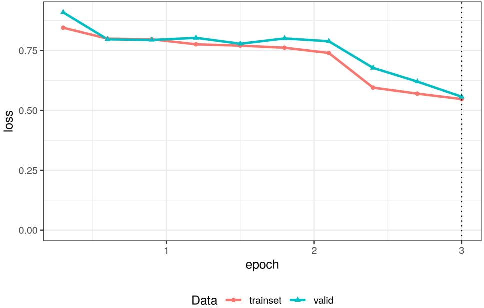


Just right:

- model has learned
- saturation before end
- no overlearning

→ this is what we want the curves to look like

3/ Fine-tuning tricks

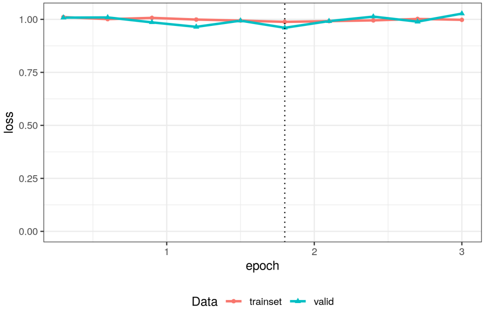


Could learn more:

- model has learned
- no saturation

$\rightarrow$ increase number of epochs (continue learning)
(or increase learning step)

3/ Fine-tuning tricks



Learns nothing:

- flat curves
- fine-tuning failed

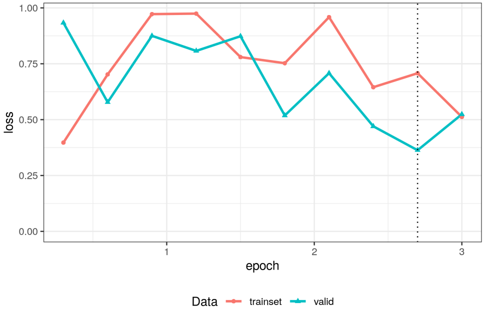
$\rightarrow$ Increase learning step

(or decrease batch size, more updates of same step)

OR

Tag more data

3/ Fine-tuning tricks

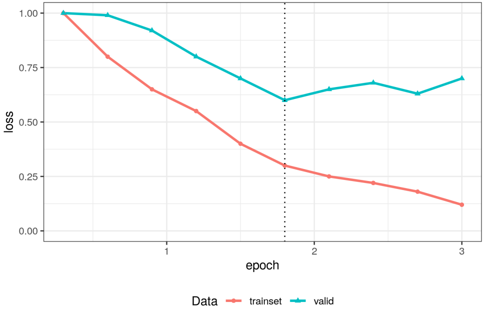


Chaotic:

- model learns something
- too much change at each step

$\rightarrow$ decrease step size (or larger batch)

3/ Fine-tuning tricks



Overlearning:

- model learns, but...
- trainset curve steeper than validation curve
- training loss significantly lower than validation loss at "best" point

→ higher weight decay (and/or lower step size)
Or tag more data

Further reading

Textbook:

- Dan Jurafsky and James H. Martin, 2023, *Speech and Language Processing* (3rd ed. draft), chapter I.11, https://web.stanford.edu/~jurafsky/slp3

Applications:

- S. Do, É. Ollion, R. Shen, (2022), "The Augmented Social Scientist: Using Sequential Transfer Learning to Annotate Millions of Texts with Human-Level Accuracy". *Sociological Methods & Research*, 53(3), 1167-1200. https://doi.org/10.1177/00491241221134526
- J. Boelaert, S. Coavoux, E. Delaine, A. Despres, S. Gollac, N. Keyhani, É. Ollion (2025). "La part du genre. Genre et approche intersectionnelle dans les revues de sciences sociales françaises au XXIe siècle", *Actes de la recherche en sciences sociales* (forthcoming)

Further reading

Model articles:

- J. Devlin, M. Chang, K. Lee, and K. Toutanova, 2019, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*
- Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, 2019, "RoBERTa: A robustly optimized BERT pretraining approach", arXiv preprint arXiv:1907.11692
- P. He, X. Liu, J. Gao, and W. Chen, 2020, "Deberta: Decoding-enhanced BERT with disentangled attention", *International Conference on Learning Representations*
- P. He, J. Gao, W. Chen, 2023, "DeBERTa V3: Improving DeBERTa Using Electra-style Pre-training with Gradient-Disentangled Embedding Sharing", *ICLR conference paper*, `https://arxiv.org/abs/2111.09543`
- B. Warner *et al*, 2025, "Smarter, Better, Faster, Longer: A Modern Bidirectional Encoder for Fast, Memory Efficient, and Long Context Finetuning and Inference", `https://arxiv.org/abs/2412.13663` (ModernBERT)

Questions and comments: `julien.boelaert@univ-lille.fr`