

Domain-Specific NLP

Contents

1. Domain-Specific Models
 - a. *Don't Stop Pre-training*
 - b. Specialized Models (BioBERT, SciBERT, Galactica)
2. Unsupervised Classification Models
 - a. Document Representation
 - b. SimCSE, E5, GTE...
3. Learning Long-Range Dependencies
 - a. Long-range attention models
 - b. State-space models: S4

Domain-Specific Models

Domain-Specific Models

Pretrained (Large) Language Models are trained on content crawled over the internet, books, reports and news papers and are, hence **are open-domain**.

A **textual domain** is the **distribution over language characterizing a given topic or genre** [1].

- You are more likely to see the word "integer" in computer science than in news papers.
- An (L)LM will be more perplex to the word "integer" even though the input comes from a StackOverflow post.

Don't Stop Pretraining

Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. [1]

Don't Stop Pretraining

Table 1: List of the domain-specific unlabeled datasets. In columns 5 and 6, we report ROBERTA's masked LM loss on 50K randomly sampled held-out documents from each domain before ($\mathcal{L}_{ROB.}$) and after (\mathcal{L}_{DAPT}) *DAPT* (lower implies a better fit on the sample). ‡ indicates that the masked LM loss is estimated on data sampled from sources similar to ROBERTA's pretraining corpus.

Don't Stop Pretraining

Figure 2: Vocabulary overlap (%) between domains. PT denotes a sample from sources similar to ROBERTA's pretraining corpus. Vocabularies for each domain are created by considering the top 10K most frequent words (excluding stopwords) in documents sampled from each domain.

Don't Stop Pretraining

Table 5: Results on different phases of adaptive pretraining compared to the baseline RoBERTa (col. 1). Our approaches are *DAPT* (col. 2, §3), *TAPT* (col. 3, §4), and a combination of both (col. 4).

Don't Stop Pretraining

"We show that **pretraining the model towards a specific task or small corpus can provide significant benefits**. Our findings suggest it may be valuable to complement work on ever-larger LMs with parallel efforts to **identify and use domain and task relevant corpora to specialize models**."

BioBERT

"[..] the word distributions of general and biomedical corpora are quite different, which can often be a problem for biomedical text mining models." [2]

BioBERT

Table 1. List of text corpora used for BioBERT

BioBERT

"We showed that **pre-training BERT on biomedical corpora is crucial in applying it to the biomedical domain**. Requiring minimal task-specific architectural modification, **BioBERT outperforms previous models on biomedical text mining tasks** such as NER, RE and QA."

SciBERT

"[...] while both BERT and ELMo have released pretrained models, they are still trained on general domain corpora such as news articles and Wikipedia." [3]

SciBERT

Table 1: Test performances of all BERT variants on all tasks and datasets. [...]

SciBERT

Table 2: Comparing SciBERT with the reported BioBERT results on biomedical datasets.

SciBERT

NB: SciBERT was trained on curated textual data ; not trained on code or script for example --at least not trained directly and purposefully on this kind of data

Galactica

"Computing has indeed revolutionized how research is conducted, but information overload remains an overwhelming problem [...]. In this paper, we argue for a better way through large language models. Unlike search engines, language models can potentially store, combine and reason about scientific knowledge." [4]

- Galactica was trained on a rather small highly curated dataset.
- All the data was standardized as markdown text.

Galactica

Table 1: Tokenizing Nature. Galactica trains on text sequences that represent scientific phenomena.

Galactica

1. **Citations:** we wrap citations with special reference tokens [START_REF] and [END_REF].
2. **Step-by-Step Reasoning:** we wrap step-by-step reasoning with a working memory token , mimicking an internal working memory context.
3. **Mathematics:** for mathematical content, with or without LaTeX, we split ASCII operations into individual characters. Parentheses are treated like digits. The rest of the operations allow for unsplit repetitions. Operation characters are !"#\$%&'*+,-./:;<=>?^_`| and parentheses are ()[]{}.

4. **Numbers:** we split digits into individual tokens. For example 737612.62 -> 7,3,7,6,1,2,,6,2.
5. **SMILES formula:** we wrap sequences with [START_SMILES] and [END_SMILES] and apply characterbased tokenization. Similarly we use [START_I_SMILES] and [END_I_SMILES] where isomeric SMILES is denoted. For example, C(C(=O)O)N → C,(,C,(,=,O,),O,),N.
6. **Amino acid sequences:** we wrap sequences with [START_AMINO] and [END_AMINO] and apply character-based tokenization, treating each amino acid character as a single token. For example, MIRLGAPQTL -> M,I,R,L,G,A,P,Q,T,L.

7. **DNA sequences:** we also apply a character-based tokenization, treating each nucleotide base as a token, where the start tokens are [START_DNA] and [END_DNA]. For example, CGGTACCCTC -> C, G, G, T, A, C, C, C, T, C.

Galactica

Figure 5: Prompt Pre-training. Pre-training weighs all tokens equally as part of the self-supervised loss. This leads to a weak relative signal for tasks of interest, meaning model scale has to be large to work. Instruction tuning boosts performance post hoc, and can generalize to unseen tasks of interest, but it risks performance in tasks that are distant from instruction set tasks. Prompt pre-training has a weaker task of interest bias than instruction tuning but less risk of degrading overall task generality.

Galactica

- **GeLU Activation** - GeLU activations for all model sizes.
- **Context Window** - a 2048 length context window.
- **No Biases** - following PaLM, we do not use biases in any of the dense kernels or layer norms.
- **Learned Positional Embeddings** - learned positional embeddings for the model.
- **Vocabulary** - vocabulary of 50k tokens using BPE. The vocabulary was generated from a randomly selected 2% subset of the training data.

Galactica

Gaussian Error Linear Units function (GeLu)

$$GELU(x) = x * \Phi(x)$$

Where $\Phi(x)$ is the Gaussian function.

$$GELU(x) \approx x * \frac{1}{2} (1 + \tanh(\frac{2}{\pi} * (x + 0.044715 * x^3)))$$

Galactica



- Allows small negative values when $x < 0$.
- Avoids the dying ReLU problem.

Galactica

Why no biases?



Unsupervised Classification Models

Document Representation

[6]

Document Representation

[5]

Document Representation

[6]

Document Representation

- (1) The data is being compressed multiple times -> challenging document can be hard to embed.
- (2) Slow to process, as we need to chunk the inputs to make multiple inferences.

Can we do better?

SimCSE

Contrastive learning uses similar data point and opposite ones in order for the model build close representations for the first ones and and more separated ones for the latter. [7]

- Unsupervised SimCSE: standard dropout as data augmentation
- Supervised SimCSE: use pairs in NLI datasets

SimCSE

$$\mathcal{L}_{uns} = -\log \frac{\exp(\frac{\text{sim}(\mathbf{h}_i, \mathbf{h}_i^+)}{\tau})}{\sum_{j=1}^N \exp(\frac{\text{sim}(\mathbf{h}_i, \mathbf{h}_j^+)}{\tau})}$$

$$\mathcal{L}_{sup} = -\log \frac{\exp(\frac{\text{sim}(\mathbf{h}_i, \mathbf{h}_i^+)}{\tau})}{\sum_{j=1}^N \exp(\frac{\text{sim}(\mathbf{h}_i, \mathbf{h}_j^+)}{\tau}) + \exp(\frac{\text{sim}(\mathbf{h}_i, \mathbf{h}_j^-)}{\tau})}$$

[8]

SimCSE

SimCSE

- The pretrained embeddings are being regularized to be more uniform.
- Semantically close pairs are better aligned.

Better performances, hence solving (1).

See also [9] [10]

SimCSE

Learning Long-Range Dependencies

Long-range attention models

Sliding window attention: Longformer [11]

Long-range attention models

Sliding window attention: Mistral 7B [12]

State-space models: Mamba

Gu, Albert, and Tri Dao. “Mamba: Linear-Time Sequence Modeling with Selective State Spaces.” arXiv, December 1, 2023.

<https://doi.org/10.48550/arXiv.2312.00752>.

Questions?

References

[1] Gururangan, Suchin, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. “[Don’t Stop Pretraining: Adapt Language Models to Domains and Tasks.](#)” In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, edited by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, 8342–60. Online: Association for Computational Linguistics, 2020.

[2] Lee, Jinhyuk, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. “[BioBERT: A Pre-Trained Biomedical Language Representation Model for Biomedical Text Mining](#).” *Bioinformatics* 36, no. 4 (February 15, 2020): 1234–40.

[3] Beltagy, Iz, Kyle Lo, and Arman Cohan. “[SciBERT: A Pretrained Language Model for Scientific Text](#).” 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)*, edited by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, 3615–20. Hong Kong, China: Association for Computational Linguistics, 2019.

[4] Taylor, Ross, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. “[Galactica: A Large Language Model for Science.](#)” arXiv, November 16, 2022.

[5] Nurmambetova, Elvira, et al. "Developing an Inpatient Electronic Medical Record Phenotype for Hospital-Acquired Pressure Injuries: Case Study Using Natural Language Processing Models." JMIR AI 2.1 (2023): e41264.

[6] Reimers, Nils, and Iryna Gurevych. “[Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks](#).” 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), edited by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, 3982–92. Hong Kong, China: Association for Computational Linguistics, 2019.

[7] Gao, Tianyu, Xingcheng Yao, and Danqi Chen. “[SimCSE: Simple Contrastive Learning of Sentence Embeddings](#).” In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, edited by Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, 6894–6910. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, 2021.

[8] Oord, Aaron van den, Yazhe Li, and Oriol Vinyals. “[Representation Learning with Contrastive Predictive Coding](#).” arXiv, January 22, 2019.

[9] Wang, Liang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. “[Text Embeddings by Weakly-Supervised Contrastive Pre-Training](#).” arXiv, December 7, 2022.

[10] Li, Zehan, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. “[Towards General Text Embeddings with Multi-Stage Contrastive Learning](#).” arXiv, August 6, 2023.

[11] Beltagy, Iz, Matthew E. Peters, and Arman Cohan. “[Longformer: The Long-Document Transformer](#).” arXiv, December 2, 2020.

[12] Jiang, Albert Q., Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, et al. “[Mistral 7B.](#)” arXiv, October 10, 2023.