

Table of Contents

Preface	xxv
Chapter 1: What are Transformers?	1
The ecosystem of transformers	3
Industry 4.0 • 3	
Foundation models • 4	
<i>Is programming becoming a sub-domain of NLP?</i> • 6	
<i>The future of artificial intelligence specialists</i> • 8	
Optimizing NLP models with transformers	9
The background of transformers • 10	
What resources should we use?	11
The rise of Transformer 4.0 seamless APIs • 12	
Choosing ready-to-use API-driven libraries • 14	
Choosing a Transformer Model • 16	
The role of Industry 4.0 artificial intelligence specialists • 16	
Summary	18
Questions	19
References	19
Chapter 2: Getting Started with the Architecture of the Transformer Model	21
The rise of the Transformer: Attention is All You Need	22
The encoder stack • 24	

<i>Input embedding</i> • 26	
<i>Positional encoding</i> • 29	
<i>Sublayer 1: Multi-head attention</i> • 36	
<i>Sublayer 2: Feedforward network</i> • 53	
The decoder stack • 54	
<i>Output embedding and position encoding</i> • 55	
<i>The attention layers</i> • 55	
<i>The FFN sublayer, the post-LN, and the linear layer</i> • 56	
Training and performance	57
Tranformer models in Hugging Face	57
Summary	58
Questions	59
References	60
 Chapter 3: Fine-Tuning BERT Models	 61
The architecture of BERT	62
The encoder stack • 62	
<i>Preparing the pretraining input environment</i> • 65	
<i>Pretraining and fine-tuning a BERT model</i> • 68	
Fine-tuning BERT	71
Hardware constraints • 71	
Installing the Hugging Face PyTorch interface for BERT • 71	
Importing the modules • 72	
Specifying CUDA as the device for torch • 72	
Loading the dataset • 73	
Creating sentences, label lists, and adding BERT tokens • 75	
Activating the BERT tokenizer • 75	
Processing the data • 76	
Creating attention masks • 76	
Splitting the data into training and validation sets • 76	
Converting all the data into torch tensors • 77	

Selecting a batch size and creating an iterator • 77	
BERT model configuration • 78	
Loading the Hugging Face BERT uncased base model • 80	
Optimizer grouped parameters • 82	
The hyperparameters for the training loop • 83	
The training loop • 83	
Training evaluation • 85	
Predicting and evaluating using the holdout dataset • 86	
Evaluating using the Matthews Correlation Coefficient • 87	
The scores of individual batches • 88	
Matthews evaluation for the whole dataset • 89	
Summary	89
Questions	90
References	90
Chapter 4: Pretraining a RoBERTa Model from Scratch	93
Training a tokenizer and pretraining a transformer	94
Building KantaiBERT from scratch	96
Step 1: Loading the dataset • 96	
Step 2: Installing Hugging Face transformers • 97	
Step 3: Training a tokenizer • 98	
Step 4: Saving the files to disk • 100	
Step 5: Loading the trained tokenizer files • 102	
Step 6: Checking resource constraints: GPU and CUDA • 103	
Step 7: Defining the configuration of the model • 104	
Step 8: Reloading the tokenizer in transformers • 104	
Step 9: Initializing a model from scratch • 105	
<i>Exploring the parameters • 106</i>	
Step 10: Building the dataset • 110	
Step 11: Defining a data collator • 111	
Step 12: Initializing the trainer • 111	

Step 13: Pretraining the model • 112	
Step 14: Saving the final model (+tokenizer + config) to disk • 113	
Step 15: Language modeling with FillMaskPipeline • 113	
Next steps	115
Summary	115
Questions	116
References	116
Chapter 5: Downstream NLP Tasks with Transformers	119
Transduction and the inductive inheritance of transformers	120
The human intelligence stack • 121	
The machine intelligence stack • 123	
Transformer performances versus Human Baselines	124
Evaluating models with metrics • 124	
Accuracy score • 125	
F1-score • 125	
Matthews Correlation Coefficient (MCC) • 126	
Benchmark tasks and datasets • 126	
From GLUE to SuperGLUE • 126	
Introducing higher Human Baselines standards • 128	
The SuperGLUE evaluation process • 129	
Defining the SuperGLUE benchmark tasks • 132	
BoolQ • 133	
Commitment Bank (CB) • 133	
Multi-Sentence Reading Comprehension (MultiRC) • 133	
Reading Comprehension with Commonsense Reasoning Dataset (ReCoRD) • 135	
Recognizing Textual Entailment (RTE) • 136	
Words in Context (WiC) • 136	
The Winograd schema challenge (WSC) • 137	
Running downstream tasks	138
The Corpus of Linguistic Acceptability (CoLA) • 138	

Stanford Sentiment TreeBank (SST-2) • 139	
Microsoft Research Paraphrase Corpus (MRPC) • 140	
Winograd schemas • 141	
Summary	142
Questions	143
References	143
Chapter 6: Machine Translation with the Transformer	145
Defining machine translation	146
Human transductions and translations • 147	
Machine transductions and translations • 148	
Preprocessing a WMT dataset	148
Preprocessing the raw data • 149	
Finalizing the preprocessing of the datasets • 152	
Evaluating machine translation with BLEU	155
Geometric evaluations • 156	
Applying a smoothing technique • 158	
Chenchermy smoothing • 159	
Translation with Google Translate	160
Translations with Trax	161
Installing Trax • 161	
Creating the original Transformer model • 162	
Initializing the model using pretrained weights • 162	
Tokenizing a sentence • 163	
Decoding from the Transformer • 163	
De-tokenizing and displaying the translation • 163	
Summary	164
Questions	165
References	166

Chapter 7: The Rise of Suprahuman Transformers with GPT-3 Engines	167
Suprahuman NLP with GPT-3 transformer models	169
The architecture of OpenAI GPT transformer models	169
The rise of billion-parameter transformer models • 170	
The increasing size of transformer models • 170	
Context size and maximum path length • 171	
From fine-tuning to zero-shot models • 173	
Stacking decoder layers • 175	
GPT-3 engines • 176	
Generic text completion with GPT-2	177
Step 9: Interacting with GPT-2 • 177	
Training a custom GPT-2 language model	179
Step 12: Interactive context and completion examples • 179	
Running OpenAI GPT-3 tasks	181
Running NLP tasks online • 181	
Getting started with GPT-3 engines • 184	
Running our first NLP task with GPT-3 • 184	
NLP tasks and examples • 187	
Comparing the output of GPT-2 and GPT-3	191
Fine-tuning GPT-3	192
Preparing the data • 193	
Step 1: Installing OpenAI • 193	
Step 2: Entering the API key • 193	
Step 3: Activating OpenAI's data preparation module • 193	
Fine-tuning GPT-3 • 194	
Step 4: Creating an OS environment • 194	
Step 5: Fine-tuning OpenAI's Ada engine • 194	
Step 6: Interacting with the fine-tuned model • 195	
The role of an Industry 4.0 AI specialist	195
Initial conclusions • 196	

Summary	197
Questions	198
References	198
 Chapter 8: Applying Transformers to Legal and Financial Documents for AI Text Summarization	 201
 Designing a universal text-to-text model	 202
The rise of text-to-text transformer models •	203
A prefix instead of task-specific formats •	204
The T5 model •	206
 Text summarization with T5	 207
Hugging Face •	207
Hugging Face transformer resources •	208
Initializing the T5-large transformer model •	210
Getting started with T5 •	210
Exploring the architecture of the T5 model •	212
Summarizing documents with T5-large •	214
Creating a summarization function •	215
A general topic sample •	216
The Bill of Rights sample •	217
A corporate law sample •	219
 Summarization with GPT-3	 220
Summary	222
Questions	223
References	223
 Chapter 9: Matching Tokenizers and Datasets	 225
 Matching datasets and tokenizers	 226
Best practices •	226
Step 1: Preprocessing •	228
Step 2: Quality control •	229

<i>Continuous human quality control</i> • 230	
Word2Vec tokenization • 232	
<i>Case 0: Words in the dataset and the dictionary</i> • 235	
<i>Case 1: Words not in the dataset or the dictionary</i> • 236	
<i>Case 2: Noisy relationships</i> • 239	
<i>Case 3: Words in the text but not in the dictionary</i> • 239	
<i>Case 4: Rare words</i> • 240	
<i>Case 5: Replacing rare words</i> • 241	
<i>Case 6: Entailment</i> • 242	
Standard NLP tasks with specific vocabulary	243
Generating unconditional samples with GPT-2 • 243	
Generating trained conditional samples • 246	
Controlling tokenized data • 247	
Exploring the scope of GPT-3	250
Summary	252
Questions	252
References	253
 Chapter 10: Semantic Role Labeling with BERT-Based Transformers	 255
Getting started with SRL	256
Defining semantic role labeling • 256	
<i>Visualizing SRL</i> • 257	
Running a pretrained BERT-based model • 258	
<i>The architecture of the BERT-based model</i> • 258	
<i>Setting up the BERT SRL environment</i> • 259	
SRL experiments with the BERT-based model	259
Basic samples	261
Sample 1 • 261	
Sample 2 • 263	
Sample 3 • 266	

Difficult samples	267
Sample 4 • 268	
Sample 5 • 270	
Sample 6 • 272	
Questioning the scope of SRL	273
The limit of predicate analysis • 274	
Redefining SRL • 276	
Summary	277
Questions	278
References	279
 Chapter 11: Let Your Data Do the Talking: Story, Questions, and Answers	 281
Methodology	283
Transformers and methods • 283	
Method 0: Trial and error	284
Method 1: NER first	287
Using NER to find questions • 287	
Location entity questions • 289	
Person entity questions • 293	
Method 2: SRL first	293
Question-answering with ELECTRA • 295	
Project management constraints • 297	
Using SRL to find questions • 298	
Next steps	303
Exploring Haystack with a RoBERTa model • 304	
Exploring Q&A with a GTP-3 engine • 305	
Summary	306
Questions	307
References	308

Chapter 12: Detecting Customer Emotions to Make Predictions	309
Getting started: Sentiment analysis transformers	310
The Stanford Sentiment Treebank (SST)	310
Sentiment analysis with RoBERTa-large • 314	
Predicting customer behavior with sentiment analysis	316
Sentiment analysis with DistilBERT • 316	
Sentiment analysis with Hugging Face's models' list • 319	
<i>DistilBERT for SST</i> • 320	
<i>MiniLM-L12-H384-uncased</i> • 322	
<i>RoBERTa-large-mnli</i> • 323	
<i>BERT-base multilingual model</i> • 324	
Sentiment analysis with GPT-3	326
Some Pragmatic I4.0 thinking before we leave	327
Investigating with SRL • 328	
Investigating with Hugging Face • 330	
Investigating with the GPT-3 playground • 332	
<i>GPT-3 code</i> • 333	
Summary	335
Questions	336
References	336
Chapter 13: Analyzing Fake News with Transformers	339
Emotional reactions to fake news	340
Cognitive dissonance triggers emotional reactions • 341	
<i>Analyzing a conflictual Tweet</i> • 342	
<i>Behavioral representation of fake news</i> • 345	
A rational approach to fake news	347
Defining a fake news resolution roadmap • 348	
The gun control debate • 349	
<i>Sentiment analysis</i> • 350	

Named entity recognition (NER) • 352	
Semantic Role Labeling (SRL) • 354	
Gun control SRL • 356	
Reference sites • 357	
COVID-19 and former President Trump's Tweets • 361	
Semantic Role Labeling (SRL) • 361	
Before we go	364
Summary	364
Questions	365
References	365
Chapter 14: Interpreting Black Box Transformer Models	367
Transformer visualization with BertViz	368
Running BertViz • 368	
Step 1: Installing BertViz and importing the modules • 369	
Step 2: Load the models and retrieve attention • 369	
Step 3: Head view • 370	
Step 4: Processing and displaying attention heads • 370	
Step 5: Model view • 372	
LIT	374
PCA • 374	
Running LIT • 375	
Transformer visualization via dictionary learning	377
Transformer factors • 377	
Introducing LIME • 379	
The visualization interface • 380	
Exploring models we cannot access	384
Summary	385
Questions	386
References	386

Chapter 15: From NLP to Task-Agnostic Transformer Models	389
Choosing a model and an ecosystem	390
The Reformer	392
Running an example • 393	
DeBERTa	395
Running an example • 395	
From Task-Agnostic Models to Vision Transformers	396
ViT – Vision Transformers • 397	
<i>The Basic Architecture of ViT • 398</i>	
<i>Vision transformers in code • 400</i>	
CLIP • 402	
<i>The Basic Architecture of CLIP • 403</i>	
<i>CLIP in code • 403</i>	
DALL-E • 408	
<i>The Basic Architecture of DALL-E • 408</i>	
<i>DALL-E in code • 409</i>	
An expanding universe of models	412
Summary	414
Questions	414
References	415
Chapter 16: The Emergence of Transformer-Driven Copilots	417
Prompt engineering	419
Casual English with a meaningful context • 420	
Casual English with a metonymy • 422	
Casual English with an ellipsis • 422	
Casual English with vague context • 423	
Casual English with sensors • 424	
Casual English with sensors but no visible context • 425	
Formal English conversation with no context • 425	

Prompt engineering training • 426	
Copilots 426	
GitHub Copilot • 426	
Domain-specific GPT-3 engines 430	
Embedding2ML • 430	
Step 1: Installing and importing OpenAI • 431	
Step 2: Loading the dataset • 432	
Step 3: Combining the columns • 433	
Step 4: Running the GPT-3 embedding • 433	
Step 5: Clustering (k-means clustering) with the embeddings • 435	
Step 6: Visualizing the clusters (t-SNE) • 436	
Instruct series • 437	
Content filter • 438	
Transformer-based recommender systems 440	
General-purpose sequences • 440	
Dataset pipeline simulation with RL using an MDP • 442	
Training customer behaviors with an MDP • 443	
Simulating consumer behavior with an MDP • 445	
Making recommendations • 445	
Computer vision 447	
Humans and AI copilots in metaverses 450	
From looking at to being in • 451	
Summary 451	
Questions 452	
References 452	
 Chapter 17: The Consolidation of Suprahuman Transformers with OpenAI's ChatGPT and GPT-4 455	
Consolidating suprahuman NLP with ChatGPT and GPT-4 transformer models 456	
How to get the most out of this chapter • 457	
Build on your knowledge • 457	

<i>Focus on the innovations • 457</i>	
<i>GitHub repository • 457</i>	
<i>Accepting the limitations of a disruptive innovation • 457</i>	
Who owns AI-created content? • 458	
<i>Opportunities • 459</i>	
<i>List of models in this chapter • 460</i>	
Jump-starting the ChatGPT API	462
Content generation • 462	
<i>Response • 463</i>	
<i>Extracting the text of the response • 464</i>	
ChatGPT Plus writes and comments on a program	464
Designing the prompt • 465	
ChatGPT Plus writes the program • 466	
ChatGPT Plus plots the output • 468	
Getting started with the GPT-4 API	470
1. GPT-4 helps to explain how to write source code • 470	
2. GPT-4 creates a function to show the YouTube presentation of GPT-4 by Greg Brockman on March 14th, 2023 • 470	
3. GPT-4 creates an application for WikiArt to display images • 471	
4. GPT-4 creates an application to display IMDb reviews • 472	
5. GPT-4 creates an application to display a news feed • 473	
6. GPT-4 creates a k-means clustering (kmc) algorithm • 474	
7. Talking with GPT-4 about GPT-4 and GPT model architecture • 475	
Advanced prompt engineering	476
1. Building a knowledge base for ChatGPT and GPT-4 • 477	
2. Adding keywords and parsing user requests • 478	
3. Building prompts that pilot ChatGPT • 478	
4. Moderation, quality control • 479	
Explainable AI (XAI)	480
Speech-to-text with Whisper • 480	
Text-to-speech with gTTS • 481	

Text-to-speech with gTTS and IPython • 483	
Transcribing audio files with Whisper • 483	
Getting started with the DALL-E 2 API	484
Creating a new image • 485	
Creating a variation of an image • 487	
Putting it all together	487
Summary	489
Questions	490
References	490
Appendix I — Terminology of Transformer Models	491
Stack	492
Sublayer	492
Attention heads	493
Appendix II — Hardware Constraints for Transformer Models	495
The Architecture and Scale of Transformers	495
Why GPUs are so special	496
GPUs are designed for parallel computing	497
GPUs are also designed for matrix multiplication	497
Implementing GPUs in code	498
Testing GPUs with Google Colab	499
Google Colab Free with a CPU	499
Google Colab Free with a GPU • 500	
Google Colab Pro with a GPU	502
Appendix III — Generic Text Completion with GPT-2	505
Step 1: Activating the GPU	506
Step 2: Cloning the OpenAI GPT-2 repository	506
Step 3: Installing the requirements	508

Step 4: Checking the version of TensorFlow	508
Step 5: Downloading the 345M-parameter GPT-2 model	509
Steps 6-7: Intermediate instructions	511
Steps 7b-8: Importing and defining the model	512
Step 9: Interacting with GPT-2	514
References	515

Appendix IV — Custom Text Completion with GPT-2 **517**

Training a GPT-2 language model	517
Step 1: Prerequisites • 517	
Steps 2 to 6: Initial steps of the training process • 518	
Step 7: The N Shepperd training files • 519	
Step 8: Encoding the dataset • 520	
Step 9: Training a GPT-2 model • 521	
Step 10: Creating a training model directory • 522	
Step 11: Generating unconditional samples • 523	
Step 12: Interactive context and completion examples • 524	
References	527

Appendix V — Answers to the Questions **529**

Chapter 1, What are Transformers?	529
Chapter 2, Getting Started with the Architecture of the Transformer Model	530
Chapter 3, Fine-Tuning BERT Models	531
Chapter 4, Pretraining a RoBERTa Model from Scratch	532
Chapter 5, Downstream NLP Tasks with Transformers	533
Chapter 6, Machine Translation with the Transformer	534
Chapter 7, The Rise of Suprahuman Transformers with GPT-3 Engines	535
Chapter 8, Applying Transformers to Legal and Financial Documents for AI Text	
Summarization	536

Chapter 9, Matching Tokenizers and Datasets	537
Chapter 10, Semantic Role Labeling with BERT-Based Transformers	538
Chapter 11, Let Your Data Do the Talking: Story, Questions, and Answers	539
Chapter 12, Detecting Customer Emotions to Make Predictions	540
Chapter 13, Analyzing Fake News with Transformers	541
Chapter 14, Interpreting Black Box Transformer Models	542
Chapter 15, From NLP to Task-Agnostic Transformer Models	543
Chapter 16, The Emergence of Transformer-Driven Copilots	544
Chapter 17, The Consolidation of Suprahuman Transformers with OpenAI's ChatGPT and GPT-4	545
Other Books You May Enjoy	549
Index	553