生成AIパスポート 対策テキスト

第2章 生成AI ①仕組み編

CNN・VAE・GAN・RNN・LSTM・Transformer を1本の線でつなぐ。

生成AIとは／生成モデルのはじまり／画像の系譜／文章の系譜／Transformerからの枝分かれ

重要用語 32語・確認問題 9問・図解 13点

02

AI資格ラボ

YouTubeチャンネル登録者への配布テキストです。

動画「【生成AIパスポート】第2章 生成AI ①仕組み編」と対応しています。

はじめに この本の使い方

この本は、YouTubeチャンネル「AI資格ラボ」の動画「**第2章 生成AI ①仕組み編**」に対応した、配布用のテキストです。動画を見ていない方でも、これ一冊で第2章の前半が最後まで読み切れるように書いてあります。

この本が扱う範囲

生成AIパスポートの第2章は、公式シラバス（出題範囲表）で**62語**あります。第1章の39語の1.6倍で、1回で扱うには多すぎます。そこで**2冊に分けました**。

	扱う範囲	語数
①	生成AIの誕生まで（この本）	32語
②	ChatGPT（次の本）	30語

この本は**①の32語**を扱います。生成モデルがどう生まれて、どう進化して、**Transformer**にたどり着いたか。そこまでが範囲です。

⇒ **第1章でやった言葉は、やり直しません。**ニューラルネットワーク、隠れ層、ディープラーニング、特徴量。このあたりは第1章のテキストと動画にあります。まだの方は先にそちらを読んでください。

この本の構成

- **本文**……ふつうに読める文章です。まずここを読んでください。
- **試験ポイント**（青い枠）……そのまま問題になる所です。試験直前はここだけ拾い読みできます。
- ⚠**マークの枠**（赤い枠）……**間違えやすい所・引っかけが仕込まれる所**です。
- **用語**（紫の帯）……シラバスに名前が載っている用語です。全部で32語あります。
- **確認問題**（緑の枠）……テーマの区切りに3問ずつ、合計9問あります。
- **チェックリスト**（巻末）……32語を自分で説明できるか、最後に確かめられます。

おすすめの進め方

1. まず**本文を最後まで通しで読む**。分からない所で止まらず、飛ばして進んでかまいません。
2. 次に**確認問題9問**を、本文を見ずに解く。ここで落ちた所が、あなたの弱点です。
3. 落ちた所だけ本文に戻り、**自分の言葉で言い直す**。読むより言い直したほうが残ります。
4. 試験直前は、**試験ポイントの枠と、巻末の用語集だけ**を見返す。

第2章の前半は、どういう章か

先に言っておきます。この章は、**アルファベットが襲ってきます**。CNN、VAE、GAN、RNN、LSTM、Transformer。名前だけ並べられると、それだけで帰りたくなります。

ただ、この6つには**共通する事情**があります。**全部が「前のやつの弱点をつぶした結果」**なのです。順番に出てきたことには理由があります。

だから、**理由で覚えると6つが1本の線になります。**逆に、名前だけで覚えようとすると**確実に混ざります**。この本は、その線を引くために書いてあります。

この本・動画・練習問題の回し方

この本は**単体でも読めます**が、動画と練習問題と合わせると回転が速くなります。おすすめは次の4ステップです。

1. **この本（章ごとのPDF）を読む。**無料です。
2. **その章の動画を見る。**本と同じ順番・同じ図で説明しています。
3. **ホームページに戻って、その章の練習問題を解く。**登録もお金も要りません。
4. **間違えた問題の解説から、動画のその場面へ飛ぶ。**「どこで説明していたっけ」と探さずに済みます。

解けるようになったら、次の章のPDFを落として同じことを繰り返します。**練習問題は全5章ぶん、模擬試験まで用意してあります**ので、動画が追いつく前でも先に進めます。

⇨ ホームページのリンクは、動画の**概要欄**にあります。

⇨ 印刷するときは**A4・実物大（100%）**で。図の中の文字まで読める大きさで作ってあります。

目次 この本の中身

動画と合わせて使う

- この本は、YouTube「**AI資格ラボ**」の動画「**【生成AIパスポート】第2章 生成AI ①仕組み編**」と同じ順番・同じ図で作ってあります。
- **動画で流れをつかみ、この本で確かめる**。あるいは**この本を先に読んでから動画を見る**。どちらでも構いません。
- 印刷は**A4・実物大（100%）**で。図の中の文字まで読める大きさで作ってあります。

第2章① この章の全体像

最初に、これから通る道を1枚で見ておきます。細かい所は分からなくて構いません。**6つの名前が縦に並んでいる理由**だけ、頭に入れてください。

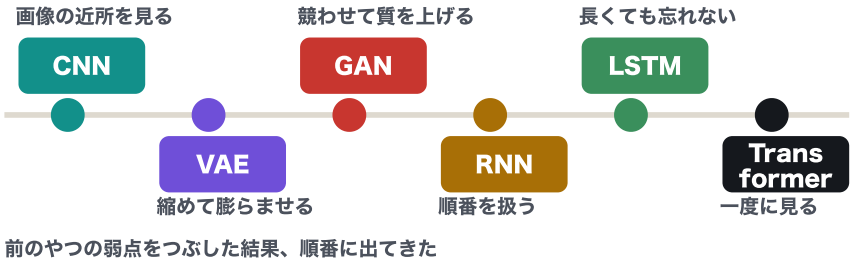

図2-1　6つは順番に出てきた。それぞれが「前のやつの弱点」をつぶしている

左から右へ、時代が進みます。**CNN**は画像の近所を見る仕掛け、**VAE**は縮めて膨らませる作り方、**GAN**は競わせる作り方、**RNN**は順番を扱う仕掛け、**LSTM**は長くても忘れない改良、そして**Transformer**が「順番に読む」こと自体をやめました。

この本は、この線を左から順にたどります。テーマの分け方は次のとおりです。

	テーマ	ここで問われること
2-1	生成AIとは何か	従来のAIとの違い。答え方は一つ
2-2	生成モデルのはじまり	ボルツマンマシンの立ち位置、何を制約したか
2-3	画像の系譜	畳み込みとは何か、VAEとGANの違い
2-4	文章の系譜	RNNの弱点、Transformerが得たものと失ったもの
2-5	枝分かれ	GPTとBERTの役割の違い、MLMとNSP

この章の勉強のコツ

- **名前を覚えるのではなく、順番と理由を覚える。**「前のやつの何が不満だったのか」が言えれば、名前は後からついてきます。
- **画像の系譜と文章の系譜は、別の線として覚える。**途中で合流するのはTransformerからです。
- **似た名前は必ず対で覚える。**エンコーダとデコーダ、生成器と識別器、MLMとNSP、RoBERTaとALBERT。試験はこの対を入れ替えて出してきます。

名前の読み方

この章はアルファベットの略語が続きます。読み方が分からないまま進むと頭に入りにくいので、先に確かめておきます。

書き方	読み方	日本語の名前
CNN	シー・エヌ・エヌ	畳み込みニューラルネットワーク
VAE	ブイ・エー・イー	変分自己符号化器
GAN	ガン	敵対的生成ネットワーク
RNN	アール・エヌ・エヌ	回帰型ニューラルネットワーク
LSTM	エル・エス・ティー・エム	長・短期記憶
Transformer	トランスフォーマー	（日本語名はなし）

⇒ **GANだけ「ガン」と一続きで読みます。**ほかは1文字ずつです。

2-1 生成AIとは何か

まず言葉から入ります。**生成AI**。英語で**ジェネレーティブAI**。シラバスにはこの両方の書き方で載っているので、**どちらで出ても同じもの**だと思ってください。

> **生成AI（ジェネレーティブAI）** Generative AI
> 文章・画像・音声・動画など、**もとは無かったものを作り出すAI**。従来の「見分けるAI」に対する呼び方。

識別するか、生成するか

では、何が「生成」なのでしょうか。第1章で扱ったAIは、**見分けるAI**でした。この写真は猫か犬か。このメールは迷惑メールか。**答えを選ぶ**仕事です。

生成AIは違います。**無かったものを作ります**。文章、画像、音声、動画。選ぶのではなく、出す。ここが決定的な差です。

図2-2　従来のAIは決まった選択肢から選ぶ。生成AIは答えが無限にある

図の左を見てください。従来のAIは、**答えが決まった選択肢の中にあります**。猫か犬か、迷惑メールかどうか。選択肢の外の答えは出てきません。

右が生成AIです。出てくるのは**新しい文章、新しい画像**。**答えは無限にあります**。だから「作った」と言えるわけです。

試験ポイント

- 試験では「**従来のAIと生成AIの違いは何か**」という聞き方をされます。
- 答え方は一つで足ります。**識別するか、生成するか**。見分けるのが従来、無いものを作るのが生成です。

生成AIも、中身は機械学習

ここで一つ、勘違いされやすい所を潰しておきます。**生成AIも、中身は機械学習です。**第1章でやった「データから学ぶ」の延長線上にあります。

まったく別の生き物が急に現れたわけではありません。**学び方と出し方が変わっただけ**です。この感覚を持っておくと、このあと出てくる6つの名前が「機械学習のいろいろな作り方」として並んで見えてきます。

生成AIが作るもの

「生成」と聞くと文章を思い浮かべがちですが、扱う対象（たいしょう）は文章だけではありません。シラバスでも、生成AIは**作り出すもの全般**を指す言葉として使われています。

作るもの	たとえば
文章	質問への答え、要約、翻訳、プログラム
画像	写真のような絵、イラスト、デザイン案
音声	読み上げ、歌声、効果音
動画	短い映像、アニメーション

この本で扱うのは、このうち**仕組みがどう生まれてきたか**です。どの種類を作る場合でも、土台になっている考え方は共通しています。

試験ポイント

- **生成AI**と**ジェネレーティブAI**は**同じもの**。シラバスに両方の表記があるので、どちらで出ても迷わないこと。
- 「作り出す対象は文章だけ」という選択肢は**誤り**です。画像・音声・動画も含みます。

⚠ ⚠ **引っかけ**　「生成AIは機械学習とは別の技術である」という選択肢は**誤り**です。生成AIは機械学習の一部です。

2-2 生成モデルのはじまり

ここから、生成モデルの歴史を順にたどります。最初に出てくるのは、**1985年**に提案されたモデルです。

ボルツマンマシン

> **ボルツマンマシン** Boltzmann Machine
> 1985年に提案された生成モデルの祖先（そせん）。**データがどうやってできているかを確率で覚えよう**とした。ユニットが全部つながっているため計算量が大きく、実用にはならなかった。

名前はいかついですが、やっていることは一言で言えます。**データがどうやってできているかを、確率で覚えようとした**。それだけです。

第1章でやったニューラルネットワークは、入力から出力へ**一方通行**に信号が流れました。ボルツマンマシンは違います。**ユニットどうしが全部つながっていて、信号が行ったり来たりします。**その揺れ動きが落ち着いたところが「データらしい状態」だ、という考え方です。

ただし、ここが大事なところです。**このモデルは実用になりませんでした。**全部がつながっているので、**計算量が爆発する**のです。

制約付きボルツマンマシン

重すぎたなら、どうするか。**つながりを減らせばいい。**それが**制約付きボルツマンマシン**です。

> **制約付きボルツマンマシン** Restricted Boltzmann Machine
> ボルツマンマシンから**同じ層の中のつながりを全部切った**もの。可視層と隠れ層の2層にし、層と層の間だけをつなぐ。計算が現実的な重さになり、ディープラーニングの実用化を支えた。

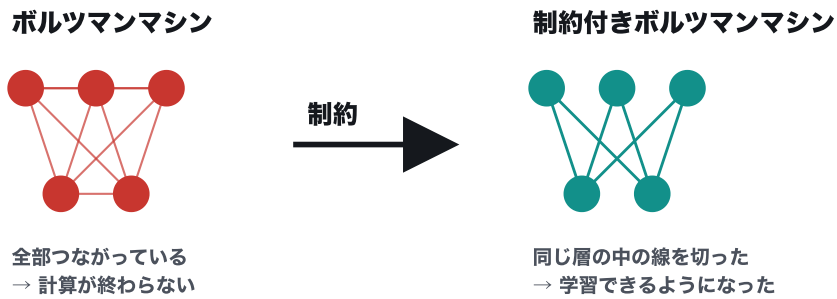

図2-3　同じ層の中の線を切っただけで、学習できるようになった

何を制約したのか。**同じ層の中のつながりを、全部切りました**。可視層と隠れ層の2層にして、**層と層の間だけをつなぐ**。層の中はつなぎません。これだけで計算が現実的な重さになりました。

そして、これが効いてくるのが**ディープラーニング**です。層を深くすると学習がうまくいかない時期が長く続いたのですが、**制約付きボルツマンマシンを積み上げて、先に軽く学習させておく**という手が使われました。つまり、**深い層を実用にした立役者のひとり**です。

試験ポイント

- **ボルツマンマシンは重い。制約をつけたら動いた。**この2文で立ち位置を覚えます。
- 試験では「**制約とは何を制約したのか**」が問われます。答えは**層の中のつながり**です。
- ボルツマンマシンは「生成モデルの祖先（そせん）」として問われます。実用になったかどうかまで含めて覚えてください。

自己回帰モデル

次は**自己回帰モデル**です。これは仕組みの名前というより、**やり方の名前**です。

> **自己回帰モデル** Autoregressive Model
> **自分が出した出力を、次の入力として使う**やり方。1つずつ順番に生成していく。GPTがこの方式。

どういうやり方か。**自分が出した答えを、次の入力にする**。それを繰り返します。文章で考えると一発で分かります。

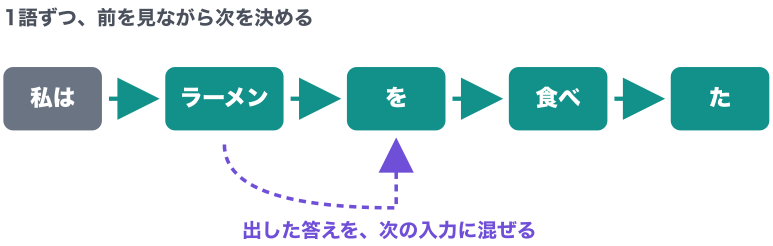

図2-4　1語出しては、それも読み込んで、また次を1語出す

「私は」と出したら、それも読み込んで次に「ラーメン」を出す。さらにそれも読み込んで「を」を出す。**1個出しては、それも読み込んで、また次を1個出す。**この繰り返しです。

……これ、どこかで見た動きではないでしょうか。**ChatGPTが、文字を1つずつ出していくあの動き。**あれです。**GPTは自己回帰モデルです。**次の本への伏線（ふくせん）ですが、ここでつないでおきます。

試験ポイント

- 問われるのは**1つずつ順番に生成する**というところです。
- 逆に言うと、**まとめて一気には出せません**。だから長い文章は時間がかかります。理屈が分かると、あの表示の遅さにも納得がいきます。

確認問題① 2-1・2-2 のふりかえり

本文を見ずに、頭の中で答えてから下の答えを見てください。

確認問題①

第1問　従来のAIと生成AIの、いちばん大きな違いは何ですか。

答

識別するか、生成するか。

見分けるのが従来のAI、無かったものを作るのが生成AIです。

確認問題①

第2問　制約付きボルツマンマシンは、何を制約したのですか。

答

同じ層の中のつながり。

層と層の間だけを残したので、計算が現実的な重さになりました。

確認問題①

第3問　自分が出した出力を次の入力に使って、1つずつ生成していくモデルを何といいますか。

答

自己回帰モデル。

GPTがこの方式です。

⇒ 3問ともできましたか。できなかった所は、いま本文に戻って**自分の言葉で言い直して**ください。読み返すより残ります。

2-3 画像の系譜

ここから**画像**の話に入ります。この節には3つの名前が出てきます。**CNN・VAE・GAN**。最初のCNNは「見分ける」側、あとの2つは「作る」側です。

CNNと畳み込み

CNN（畳み込みニューラルネットワーク） Convolutional Neural Network

畳み込みを使って画像の特徴を取り出すニューラルネットワーク。**画像を得意とする**。

畳み込み Convolution

小さな窓を画像の上でずらしながら当て、**局所的な特徴を取り出す操作**。CNNの中心となる仕組み。

第1章で、AIが画像を認識する仕組みに触れました。その主役が**CNN**、日本語で**畳み込みニューラルネットワーク**です。

何が新しかったのでしょうか。それまでは、画像の点を**全部そのまま**ニューラルネットワークに入れていました。でも画像は、**隣り合った点どうしに意味がある**のです。目の形、輪郭（りんかく）の向き、模様。**近所を見ないと分かりません。**

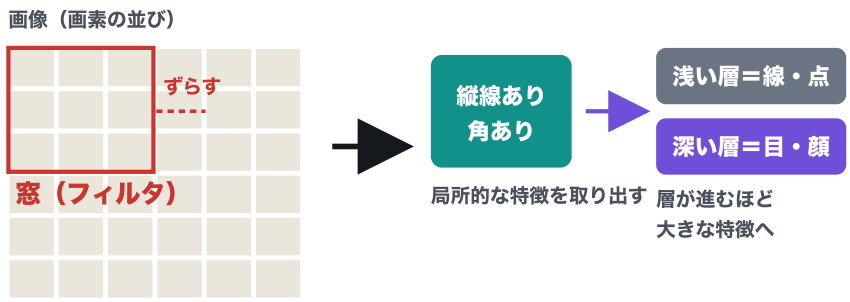

図2-5　小さな窓をずらして当て、局所的な特徴を取り出す。層が進むほど大きな特徴へ

そこで**畳み込み**です。**小さな窓を、画像の上で少しずつずらしながら当てていきます。**窓の中を見て、「ここに縦線があるな」「ここは角だな」と反応する。その反応を集めた地図を、また次の層で見る。

こうすると、**浅い層は線や点、深い層は目や顔**というふうに、だんだん大きな特徴になっていきます。第1章で**特徴量**をやりましたね。**CNNは、その特徴量を自分で見つけるための仕掛け**だと思ってください。

試験ポイント

- **畳み込みは「局所的な特徴を取り出す操作」。**この言い回しで出ます。
- **CNNは画像を得意とする。**この2つだけで十分戦えます。

⚠ ⚠ **引っかけ**　畳み込みは**画像を小さくするための操作ではありません**。あくまで**特徴を取り出す**操作です。

VAE（変分自己符号化器）

画像を**見分ける**話は済みました。ここからが**作る**話です。まず**VAE**、日本語で**変分自己符号化器**。

仕組みは、実はとても素直です。**入り口で縮めて、出口でふくらませる**。それだけです。

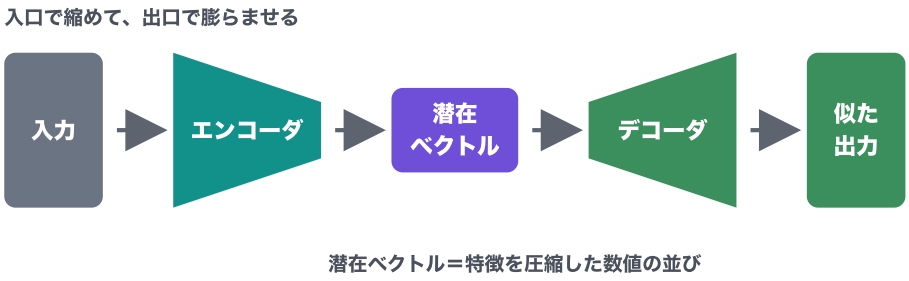

図2-6　縮める係がエンコーダ、ふくらませる係がデコーダ。間の短い数字の並びが潜在ベクトル

VAE（変分自己符号化器） Variational Auto Encoder
入力を**縮めて、揺らして、戻す**ことで新しいデータを作る生成モデル。

エンコーダ Encoder
入力を**圧縮する側**（入口）。

デコーダ Decoder
圧縮されたものから**復元する側**（出口）。

潜在ベクトル Latent Vector
エンコーダが作る**短い数字の並び**。特徴を圧縮した表現。

縮める係が**エンコーダ**、ふくらませる係が**デコーダ**。そして、縮めた結果の**短い数字の並び**が**潜在ベクトル**です。

たとえば猫の写真を、**猫らしさの成分だけに絞った数字**にします。その数字から、もう一度、猫の絵を組み立て直す。

……ここで、当然の疑問が出ます。**元に戻すだけなら、意味がないのでは？**　そのとおりです。戻すだけなら意味がありません。

ノイズ Noise

潜在ベクトルに**わざと加える揺らぎ**。これがあるので、元と違う新しいデータが出てくる。

VAEが賢(かしこ)いのはここからです。**その数字を、わざとちょっと揺らします**。この揺らぎが**ノイズ**です。少しずらした数字から組み立てると、**元の猫ではない、新しい猫が出てきます**。これが「生成」です。

試験ポイント

- 覚え方は**エンコーダで縮めて、潜在ベクトルを揺らして、デコーダで戻す**。
- **エンコーダ・潜在ベクトル・デコーダの3語はセットで出ます。**入口と出口を入れ替えた選択肢が定番です。

GAN（敵対的生成ネットワーク）

同じ「作る」でも、まったく違う発想で来たのが**GAN**です。**敵対的生成ネットワーク**。名前のとおり、**敵対**します。

GAN（敵対的生成ネットワーク） Generative Adversarial Network

生成器と識別器を競い合わせて質を上げる生成モデル。

生成器 Generator

GANの中で**偽物を作る側**。

識別器 Discriminator

GANの中で**本物か偽物かを見分ける側**。

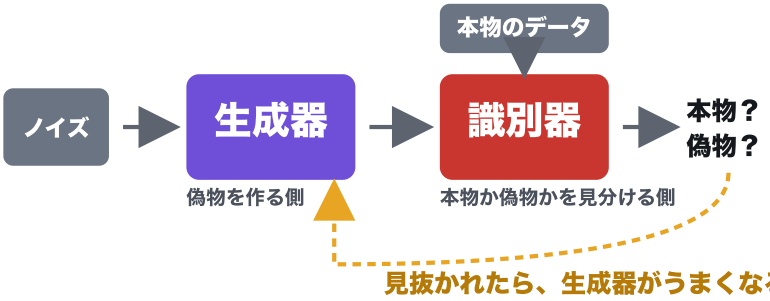

図2-7　生成器と識別器の追いかけっこ。見抜かれるほど生成器がうまくなる

中に2人います。1人目が**生成器**。**偽物を作る係**です。2人目が**識別器**。**それが本物か偽物かを見破る係**です。この2人を**戦わせます**。

生成器は「バレないように」うまくなろうとする。識別器は「見逃さないように」厳しくなろうとする。**追いかけっこをしているうちに、生成器の腕が上がっていく**。

よく偽札作りと警察にたとえられます。偽札がうまくなれば警察も目が肥える。目が肥えれば偽札もさらにうまくなる。最終的に、人間には見分けがつかない偽札ができあがる。物騒なたとえですが、試験ではこの構図がそのまま問われます。

試験ポイント

- **生成器と識別器。2つが競い合う。**この2語は必ず対で覚えます。
- VAEとの違いも聞かれます。**VAEは縮めて戻す。GANは2つを競わせる**。ここだけ押さえてください。

⚠ ⚠ **引っかけ**　「生成器と識別器が**協力して**同じ目標に向かう」という選択肢は誤りです。**競わせる**のが要点です。

確認問題② 2-3 のふりかえり

確認問題②

第1問　画像の中の、近くの点どうしの特徴を取り出す操作を何といいますか。またそれを使うネットワークは。

答

畳み込み／CNN。

畳み込みは「局所的な特徴を取り出す操作」です。

確認問題②

第2問　VAEで、入力を縮めた短い数字の並びを何といいますか。

答

潜在ベクトル。

縮めるのがエンコーダ、戻すのがデコーダです。

確認問題②

第3問　GANの中にいる2つのネットワークの名前は何ですか。

答

生成器と識別器。

偽札屋と警察、と覚えると混ざりません。

⇨ ここまでが**画像**の系譜です。次から**文章**の系譜に移ります。

2-4 文章の系譜

文章と、音声と、株価。この3つに共通するものは何でしょうか。**順番があること**です。ここからは、順番のあるデータをどう扱うかの話になります。

RNNと隠れ層・リカレント層

> **シーケンスデータ** Sequence Data
> 文章・音声・株価など、**順番に意味があるデータ**。日本語では系列データ。

こういうデータを**シーケンスデータ**、日本語で系列データといいます。「私は／ラーメンを／食べた」を並べ替えたら意味が壊(こわ)れますよね。**順番が意味を持つ**のです。

ところが、ここまでのニューラルネットワークは**入力を一度にまとめて受け取る**作りでした。順番という情報が、そもそも入りません。

> **RNN（回帰型ニューラルネットワーク）** Recurrent Neural Network
> **隠れ層の出力を次の入力に混ぜて**、順番のあるデータを扱えるようにしたモデル。

> **隠れ層** Hidden Layer
> 入力層と出力層のあいだの層。RNNでは**記憶係**の役目をする。

> **リカレント層** Recurrent Layer
> **前の状態を次へ戻す構造を持った層**。RNNの中心となる仕組み。

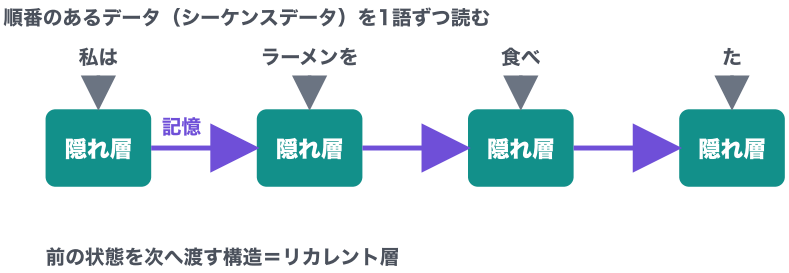

図2-8　隠れ層の出力を次へ渡す。この「戻す」構造がリカレント層

そこで出てきたのが**RNN、回帰型ニューラルネットワーク**です。何をしたか。**隠れ層の出力を、次の入力に混ぜて、もう一度同じ層に戻した**。この「戻す」構造を持った層を**リカレント層**といいます。

1語読む。覚えておく。次の語を読むときに、さっきの記憶も一緒に見る。**メモを取りながら読んでいる**イメージです。第1章で出てきた**隠れ層**、つまり入力層と出力層のあいだの層が、RNNでは**記憶係**をやっていると思ってください。

LSTM（長・短期記憶）

そのRNNですが、**弱点がありました。長い文章になると、前のほうを忘れます。**

「私は、去年の夏に、家族と一緒に、沖縄の海に行って、そこで食べたソーキそばが……」。この時点で、**主語が誰だったか怪しくなります**。なぜ忘れるかというと、記憶をぐるぐる回しているうちに**信号が薄まっていく**からです。

> **LSTM（長・短期記憶）** Long Short-Term Memory
> RNNに**忘れるかどうかを決めるスイッチ**を足したもの。長い系列でも前のほうを覚えていられる。

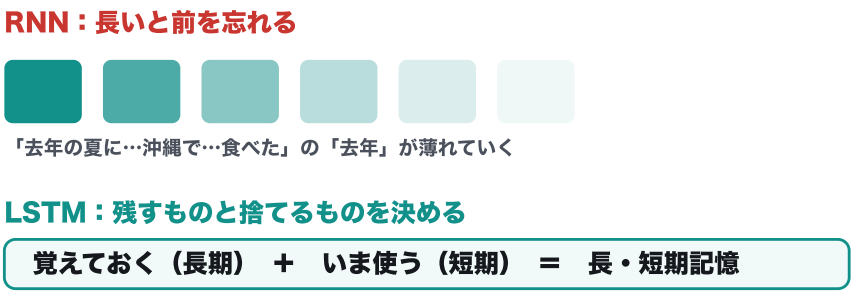

図2-9　覚えておく（長期）と、いま使う（短期）を分けて持つ

これを解決したのが**LSTM、長・短期記憶**です。何を足したか。**忘れるかどうかを決めるスイッチ**です。大事なことは長く持っておく。どうでもいいことは捨てる。このスイッチのおかげで、**長い系列でも前のほうを覚えていられる**ようになりました。

試験ポイント

- **RNNは順番を扱える。ただし長いと忘れる。忘れないようにしたのがLSTM。**この3文で足ります。
- 名前がそのまま答えです。**長・短期記憶**＝長い記憶と短い記憶を分けて持つ、と読めます。

Transformerモデル

そして、この章の主役です。**Transformerモデル。2017年**に出ました。

> **Transformerモデル** Transformer
> 順番に読むのをやめ、**文の全部の語を一度に見る**モデル。2017年に登場。いまの生成AIのほとんどがこの子孫。

何がすごかったのか。……**RNNをやめました。**雑に聞こえますが本当です。**順番に読むのを、やめたのです。**

図2-10　RNNは前が終わるまで次に進めない。Transformerは全部を一度に見る

RNNもLSTMも、**1語ずつ順番に読む**しかありませんでした。順番に読むということは、**前の語を読み終わるまで次に進めない**ということです。つまり**手分けができない**。学習にものすごく時間がかかります。

Transformerは、**文の全部の語を、一度に見ます。**そのうえで、語と語の関係を計算で出す。だから**手分けができる**。大量のデータを、現実的な時間で学習できるようになりました。

これが効きました。**いま名前を聞く生成AIは、ほぼ全部Transformerの子孫です。**GPTも、BERTも、Geminiも、Claudeも。

> **アーキテクチャ** Architecture
> **モデルの構造・設計そのもの**。建物の設計図と同じ意味。「Transformerというアーキテクチャ」のように使う。

ちなみに**アーキテクチャ**という言葉がシラバスに出てきますが、これは**モデルの構造・設計そのもの**という意味です。建物の設計図と同じ言葉で、深い意味はありません。

Attention（自己注意力）

ここからがTransformerの心臓部（しんぞうぶ）です。**Attention**、日本語だと注意機構。

シラバスには**Attention層**、**自己注意力（セルフ・アテンション）**、**Attention Mechanism** と、3つの形で載っています。まとめて**どこに注目するかを決める仕組み**だと思ってください。

> **Attention層** Attention Layer
> どの語に注目するかを**重みとして計算する層**。

> **自己注意力（Self-Attention）** Self-Attention
> **同じ文の中の語どうしで**注目し合う仕組み。自分の中でやるので「自己」。

Attention Mechanism Attention Mechanism

Attentionの仕組みそのものを指す言い方。シラバスでは上の3つが並記されているが、**指しているものは同じ**。

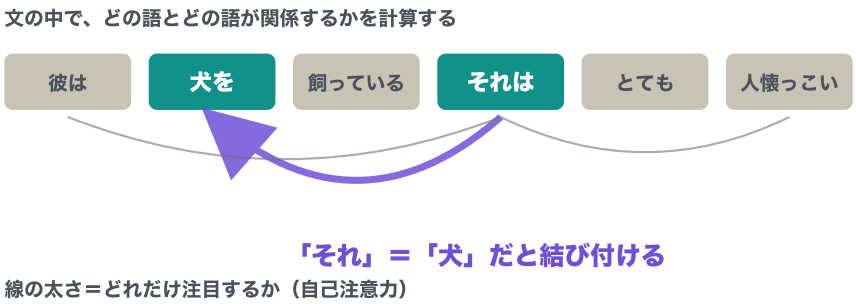

図2-11　「それ」が指すのはどれか。語と語の関係を重みで計算する

具体的にいきます。「彼は犬を飼っている。それはとても人懐っこい。」……この「それ」は何を指しますか。**犬**ですよね。「彼」ではありません。人間は当たり前にできますが、**機械にとっては難問**です。

Attentionは、「それ」を処理するときに、**文の中のどの語を強く見るか**を、**重み**として計算します。犬に強い重み、彼に弱い重み。これを**同じ文の中の語どうしでやる**のが**自己注意力**です。自分の中で注目し合うから「自己」です。

試験ポイント

- 試験では**文中のどの単語に注目すべきかを重みで決める仕組み**という言い回しで出ます。
- **Attention層・自己注意力・Attention Mechanism は同じものを指します。**3つとも覚えておけば、どの表記で出ても迷いません。

位置エンコーディング

さきほど「Transformerは文の全部の語を一度に見る」と言いました。ここで、鋭(するど)い人は気づきます。**一度に見たら、順番が分からなくないですか？**

そのとおりです。一度に見るということは、**語順の情報が消える**ということです。「犬が人を噛(か)んだ」と「人が犬を噛んだ」が区別できなくなる。これは困ります。

位置エンコーディング Positional Encoding

それぞれの語に**「何番目か」という情報を数値として足す**仕組み。Transformerが語順を扱うために必要。

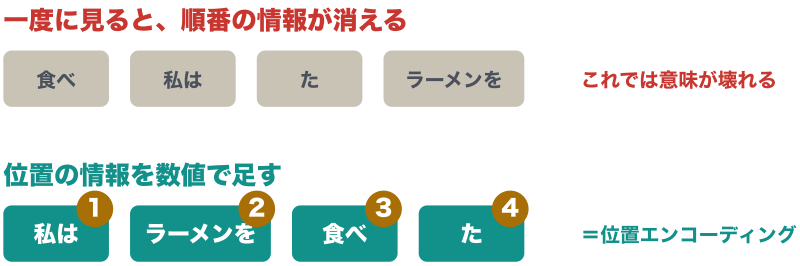

図2-12　語に「何番目です」という数値を足しておく。これが位置エンコーディング

そこで**位置エンコーディング**です。**それぞれの語に、あなたは何番目です、という情報を数字として足しておく**。これだけです。でも、これが無いと**Transformerは成立しません**。

試験ポイント

- **Transformerが語順を扱うための仕組み**として出ます。
- **Attentionとセットで覚えてください。**「一度に見る」ことで得たもの（手分け）と失ったもの（語順）が対になります。

文章の系譜を1枚で

ここまでの3つを並べます。**何ができるようになって、何が残った不満か**を横に見てください。順番に出てきた理由がそのまま表になります。

	できるようになったこと	残った不満
RNN	順番のあるデータを扱える	長いと前を忘れる
LSTM	長くても忘れない	1語ずつしか読めない＝学習が遅い
Transformer	全部を一度に見る＝手分けできる	語順が消える（位置エンコーディングで補う）

⇒ **この表の右端が、次の行の左端になっています。**「前のやつの弱点をつぶした結果」というのは、こういう意味です。

確認問題③ 2-4 のふりかえり

確認問題③

第1問　順番のあるデータを扱えるようにしたモデルは何ですか。またその弱点を直したモデルは。

答

RNN。弱点は長いと忘れること。直したのがLSTM。

RNNの記憶係が隠れ層とリカレント層です。

確認問題③

第2問　Transformerが、順番に読むのをやめたことで得たものは何ですか。

答

手分けができること。

全部の語を一度に見るので、並列に計算できて学習が速くなりました。

確認問題③

第3問　全部の語を一度に見ると失われるものは何で、それをどう補(おぎな)いましたか。

答

語順。位置エンコーディングで補った。

得たものと失ったものは対で問われます。

2-5 Transformerからの枝分かれ

Transformerができました。ここから**枝分かれ**します。片方が**創る**ほう、もう片方が**読む**ほうです。

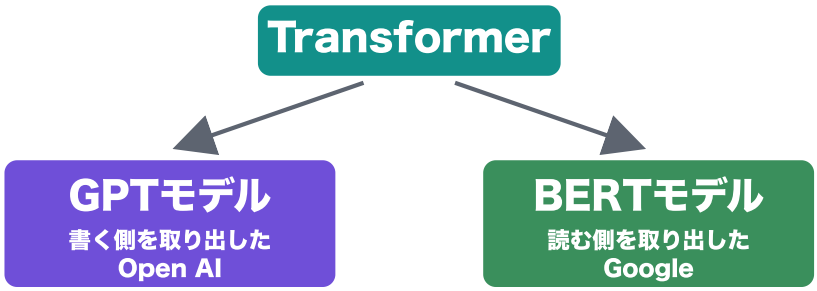

図2-13　Transformerから枝が2本。創るGPTと、読むBERT

GPTモデル（Open AI）

GPTモデル Generative Pre-trained Transformer
Transformerの**作る側**だけを取り出したモデル。**自己回帰**で前から後ろへ一方向に書いていく。

Open AI OpenAI
GPTモデルを開発した組織。

まず一方の枝が**GPTモデル**。作ったのは**Open AI**です。GPTは、**Transformerの作る側だけを取り出した**モデルです。

やっていることは、2-2でやった**自己回帰**です。**次の語を予測して、1つ出す。それも読んで、また次を出す。**これをひたすら繰り返します。

……それだけで文章が書けるのか、と思いますよね。**書けたのです。**それが今のところの答えです。覚え方は**GPTは創るほう**。前から後ろへ、**一方向に**書いていきます。

BERTモデル（MLMとNSP）

BERTモデル Bidirectional Encoder Representations from Transformers
Transformerの**読む側**を取り出したモデル。Googleが開発。**前も後ろも同時に見る**ので意味を読み取るのが得意。

もう一方の枝が**BERTモデル**です。こちらはGoogleが出しました。BERTは、**Transformerの読む側**を取り出したモデルです。

GPTが前から後ろへ一方向だったのに対して、**BERTは、前も後ろも同時に見ます。**だから**意味を読み取るのが得意**です。検索や分類に使われます。

そして、BERTの学習方法には名前がついていて、**ここが試験に出ます**。2つあります。

MLM（Masked Language Model） Masked Language Model
文の中の単語を**わざと隠して、そこに何が入るかを当てさせる**学習方法。穴埋め問題。

NSP（Next Sentence Prediction） Next Sentence Prediction
2つの文が続いているかどうかを当てさせる学習方法。続き当て。

1つ目、**MLM**。マスクド・ランゲージ・モデル。やり方は、**文の中の単語をわざと隠して、そこに何が入るかを当てさせる**。「今日は［　］がいい」→「天気」。穴埋(あなう)め問題を延々(えんえん)と解かせるわけです。前と後ろの両方を見ないと解けないので、**双方向に読む力がつきます**。

2つ目、**NSP**。ネクスト・センテンス・プリディクション。**2つの文を見せて、これは続いていますか、と当てさせます。**文と文のつながりを学ばせるためのものです。

試験ポイント

- **GPTは創る、BERTは読む**。役割の違いをまず覚えます。
- **BERTの学習は、穴埋めのMLMと、続き当てのNSP。**この2つは入れ替えて出題されます。

RoBERTaとALBERT

最後に、BERTの**改良版が2つ**シラバスに載っています。**RoBERTaとALBERT**です。細かい中身は問われません。**方向性だけ**覚えてください。

RoBERTa Robustly Optimized BERT Approach
BERTを**もっと鍛えた版**。データを増やして長く学習させ、**NSPをやめた**。

ALBERT（a Lite BERT） A Lite BERT
BERTの**軽い版**。中身を工夫してパラメータを大幅に減らし、性能を保ったまま小さくした。

RoBERTa。こちらは**もっと鍛えた版**です。データを増やして、長く学習させた。そして、**さきほどのNSPをやめました**。「続き当ては、そんなに効いていなかった」という判断です。……さっき覚えたものが、すぐ消される。AIの世界はだいたいこうです。

ALBERT。正式には**ア・ライト・バート**。名前のとおり**軽い版**です。中身を工夫して、**パラメータを大幅に減らしました**。性能を保ったまま小さくした、というのが売りです。

試験ポイント

- **RoBERTaは強くした版。ALBERTは軽くした版。**
- **名前にライトが入っているほうが軽い。**これで混ざりません。
- RoBERTaが**NSPをやめた**ことは、そのまま問題になります。

枝分かれを1枚で

	GPTモデル	BERTモデル
役割	創る（文章を書く）	読む（意味を取る）
読む向き	前から後ろへ一方向	前も後ろも同時（双方向）
作ったところ	Open AI	Google
やり方	自己回帰（1語ずつ出す）	MLM（穴埋め）とNSP（続き当て）
改良版	（次の本で扱います）	RoBERTa（強く）／ALBERT（軽く）

⚠ ⚠ **引っかけ**　**GPTとBERTの役割を入れ替えた**選択肢が定番です。**創るGPT、読むBERT**。ここだけは反射で言えるようにしてください。

まとめ 第2章① これが言えれば合格ライン

前半を、**一本の線**としてまとめます。ここが自分の言葉で言えれば、この範囲は大丈夫(だいじょうぶ)です。

はじまり

- **生成AI**は、識別ではなく**生成**をするAI。ジェネレーティブAIとも言う
- 祖先は**ボルツマンマシン**。ただし全部つながっていて重すぎた
- 層の中のつながりを切ったのが**制約付きボルツマンマシン**。これで動いた
- 1つずつ順番に出すやり方が**自己回帰モデル**。GPTがこれ

画像の系譜

- 近所を見る**畳み込み**で**CNN**。畳み込みは局所的な特徴を取り出す操作
- 作るほうは2系統。縮めて揺らして戻す**VAE**と、生成器と識別器を戦わせる**GAN**
- VAEの部品は**エンコーダ・潜在ベクトル・ノイズ・デコーダ**

文章の系譜

- 順番を扱える**RNN**。記憶係が**隠れ層**と**リカレント層**。扱うデータは**シーケンスデータ**
- 長いと忘れるので、忘却のスイッチをつけたのが**LSTM**
- そして**Transformerモデル**。順番に読むのをやめて、全部を一度に見た
- どこに注目するかを決めるのが**Attention層・自己注意力・Attention Mechanism**
- 一度に見ると消える語順を補うのが**位置エンコーディング**。この設計そのものが**アーキテクチャ**

枝分かれ

- 創る**GPTモデル**。作ったのは**Open AI**
- 読む**BERTモデル**。学習は**MLM**と**NSP**
- その改良が、強くした**RoBERTa**と、軽くした**ALBERT**

最後に

- ……**32語、全部通りました。**
- アルファベットの列に見えたものが、**弱点をつぶし続けた歴史**だったと分かれば、もう混ざりません。
- 次は**第2章の後半（ChatGPTの系譜）**です。GPT-1からGPT-5まで、数字とアルファベットがさらに増えますが、やることは同じです。

入れ替えて出される「対」

この章でいちばん多い引っかけは、**対になっている2語を入れ替える**ものです。ここだけは反射で言えるようにしてください。

対	どちらがどちら
エンコーダ／デコーダ	エンコーダが**入口で縮める**、デコーダが**出口で戻す**
生成器／識別器	生成器が**偽物を作る**、識別器が**見分ける**
VAE／GAN	VAEは**縮めて戻す**、GANは**2つを競わせる**
GPT／BERT	GPTは**創る・一方向**、BERTは**読む・双方向**
MLM／NSP	MLMは**穴埋め**、NSPは**続き当て**
RoBERTa／ALBERT	RoBERTaは**強くした版**、ALBERTは**軽くした版**

32語チェックリスト

最後に、自分の言葉で説明できるか確かめてください。**1つでも詰まったら、その語のページに戻ります。**

- [ ] 生成AI（ジェネレーティブAI）
- [ ] ボルツマンマシン
- [ ] 制約付きボルツマンマシン
- [ ] 自己回帰モデル
- [ ] CNN（畳み込みニューラルネットワーク）
- [ ] 畳み込み
- [ ] VAE（変分自己符号化器）
- [ ] ノイズ
- [ ] エンコーダ
- [ ] デコーダ
- [ ] 潜在ベクトル
- [ ] GAN（敵対的生成ネットワーク）
- [ ] 生成器
- [ ] 識別器
- [ ] RNN（回帰型ニューラルネットワーク）
- [ ] 隠れ層
- [ ] リカレント層
- [ ] シーケンスデータ
- [ ] LSTM（長・短期記憶）
- [ ] Transformerモデル
- [ ] Attention層
- [ ] 自己注意力（Self-Attention）
- [ ] Attention Mechanism
- [ ] 位置エンコーディング
- [ ] アーキテクチャ
- [ ] GPTモデル
- [ ] Open AI
- [ ] BERTモデル
- [ ] MLM（Masked Language Model）
- [ ] NSP（Next Sentence Prediction）
- [ ] RoBERTa
- [ ] ALBERT（a Lite BERT）

用語集 第2章①の重要用語 32語

公式シラバスに名前が載っている語だけを並べてあります。**p.◯ は、その語が本文で説明されているページ**です。

生成AIとそのはじまり

生成AI（ジェネレーティブAI） p.6
文章・画像・音声・動画など、無かったものを作り出すAI。従来のAIは識別、生成AIは生成。

ボルツマンマシン p.8
1985年提案。データのでき方を確率で覚えようとした生成モデルの祖先。全部つながっていて計算量が大きく実用にならなかった。

制約付きボルツマンマシン p.8
同じ層の中のつながりを切り、層と層の間だけをつないだもの。計算が現実的になり、深い層の学習を支えた。

自己回帰モデル p.9
自分が出した出力を次の入力に使い、1つずつ順番に生成していくやり方。GPTがこれ。

画像の系譜

CNN（畳み込みニューラルネットワーク） p.11
畳み込みを使って画像の特徴を取り出すネットワーク。画像を得意とする。

畳み込み p.11
小さな窓をずらしながら当て、局所的な特徴を取り出す操作。

VAE（変分自己符号化器） p.12
入力を縮めて、揺らして、戻すことで新しいデータを作る生成モデル。

ノイズ p.13
潜在ベクトルにわざと加える揺らぎ。これがあるので元と違うものが出てくる。

エンコーダ p.12
入力を圧縮する側（入口）。

デコーダ p.12
圧縮されたものから復元する側（出口）。

潜在ベクトル p.12
エンコーダが作る短い数字の並び。特徴を圧縮した表現。

GAN（敵対的生成ネットワーク） p.13
生成器と識別器を競い合わせて質を上げる生成モデル。

生成器 p.13
GANの中で偽物を作る側。

識別器 p.13
GANの中で本物か偽物かを見分ける側。

文章の系譜

RNN（回帰型ニューラルネットワーク） p.15
隠れ層の出力を次の入力に混ぜ、順番のあるデータを扱えるようにしたモデル。

隠れ層 p.15
入力層と出力層のあいだの層。RNNでは記憶係の役目をする。

リカレント層 p.15
前の状態を次へ戻す構造を持った層。

シーケンスデータ p.15
文章・音声・株価など、順番に意味があるデータ。系列データ。

LSTM（長・短期記憶） p.16
RNNに忘れるかどうかを決めるスイッチを足したもの。長い系列でも忘れにくい。

Transformerモデル p.16
2017年登場。順番に読むのをやめ、文の全部の語を一度に見るモデル。いまの生成AIのほとんどがこの子孫。

Transformerの中身

Attention層 p.17
どの語に注目するかを重みとして計算する層。

自己注意力（Self-Attention） p.17
同じ文の中の語どうしで注目し合う仕組み。

Attention Mechanism p.18
Attentionの仕組みそのものを指す言い方。上の3つは同じものを指す。

位置エンコーディング p.18
それぞれの語に「何番目か」を数値として足す仕組み。Transformerが語順を扱うために必要。

アーキテクチャ p.17
モデルの構造・設計そのもの。

枝分かれした先

GPTモデル p.21
Transformerの作る側を取り出したモデル。自己回帰で前から後ろへ一方向に書く。

Open AI p.21
GPTモデルを開発した組織。

BERTモデル p.21
Transformerの読む側を取り出したモデル。Googleが開発。前も後ろも同時に見る。

MLM（Masked Language Model） p.22
文の中の単語を隠して当てさせる学習方法。穴埋め。

NSP（Next Sentence Prediction） p.22
2つの文が続いているかを当てさせる学習方法。続き当て。

RoBERTa p.22
BERTをもっと鍛えた版。データを増やし長く学習させ、NSPをやめた。

ALBERT（a Lite BERT） p.22
BERTの軽い版。パラメータを大幅に減らし、性能を保ったまま小さくした。

この本について

生成AIパスポート 対策テキスト　第2章 生成AI ①仕組み編

発行：YouTubeチャンネル「**AI資格ラボ**」／制作：AI資格ラボ

出題範囲は**生成AIパスポート 公式シラバス（2026年2月試験より適用）**にもとづいています。最新のシラバスは主催団体の公式サイトで確認してください。

⇒ 本書は試験対策のための私家版（しかばん）テキストです。試験の実施団体とは関係がありません。内容には細心（さいしん）の注意を払っていますが、**合否を保証するものではありません**。

動画版はこちら：**YouTube「AI資格ラボ」**／第1章のテキストも同じ作りで配布しています。