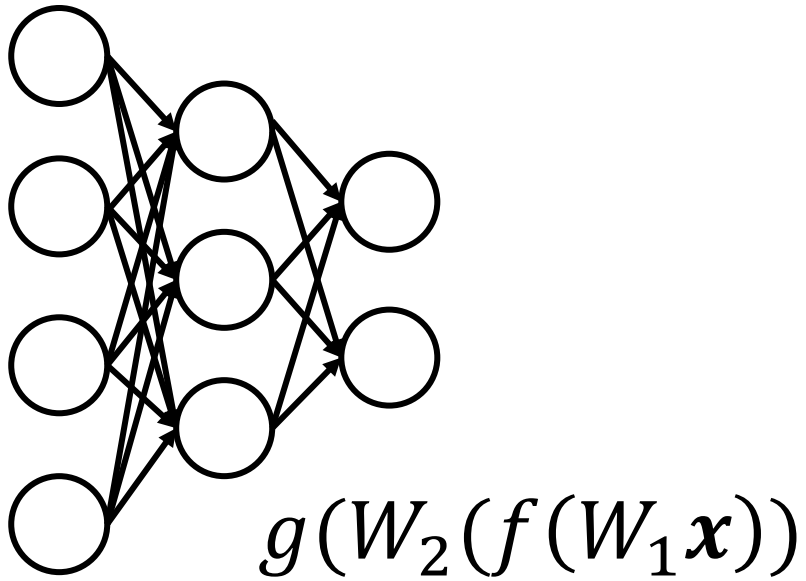
第3回
深層学習

生成AI入門

深層学習（Deep learning）

- 近年の機械学習手法のトレンド
  - 生成AIは基本的に深層学習を利用
- 深層ニューラルネットワーク (Deep neural network: DNN)
  - TransformerもDNNの一種
- ニューラルネットワーク
  - 神経系を参考にした機械学習モデル

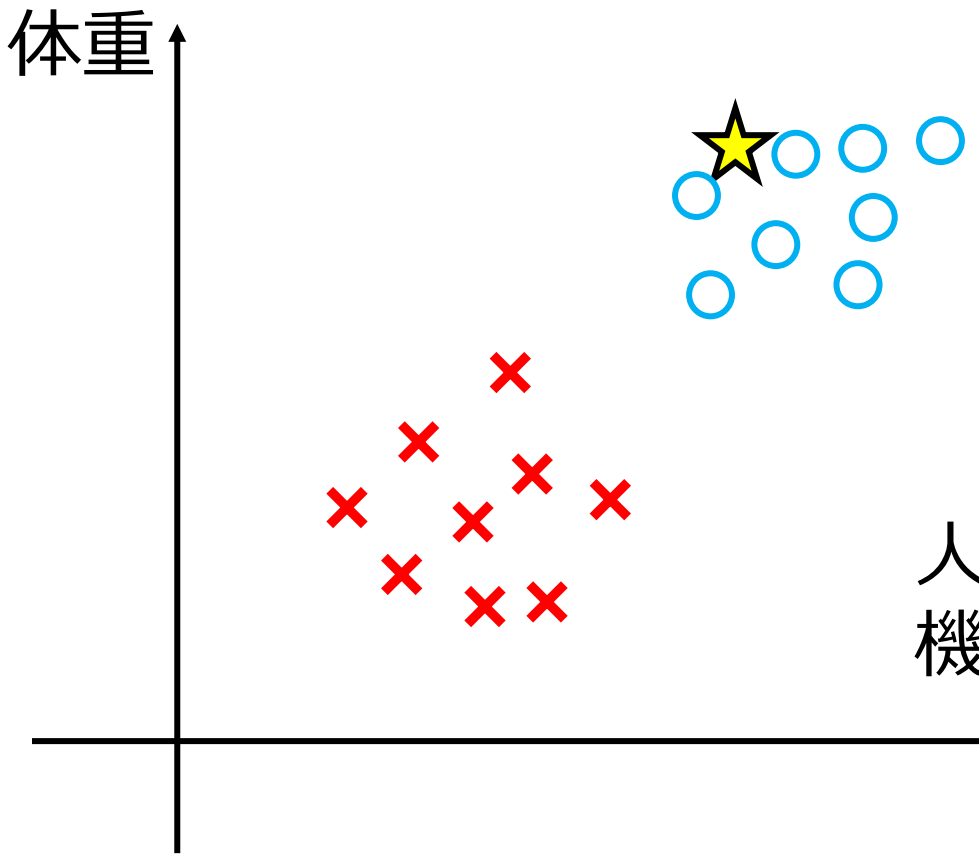


機械学習

- データをもとに，目的を達成することができるように，
  機械（モデル）が学習すること
  - 目的：「生成AI」から次の文字を推定，画像に写っているものを判定　など
  - モデル：目的を達成するための仕組みやそのために用いられる数式

- 学習
  - モデルのパラメータを調整する
  - 教師あり機械学習：「この入力には，この出力をせよ」
  という例を満たすようにパラメータを調整する

身長と体重から学年を当てる問題



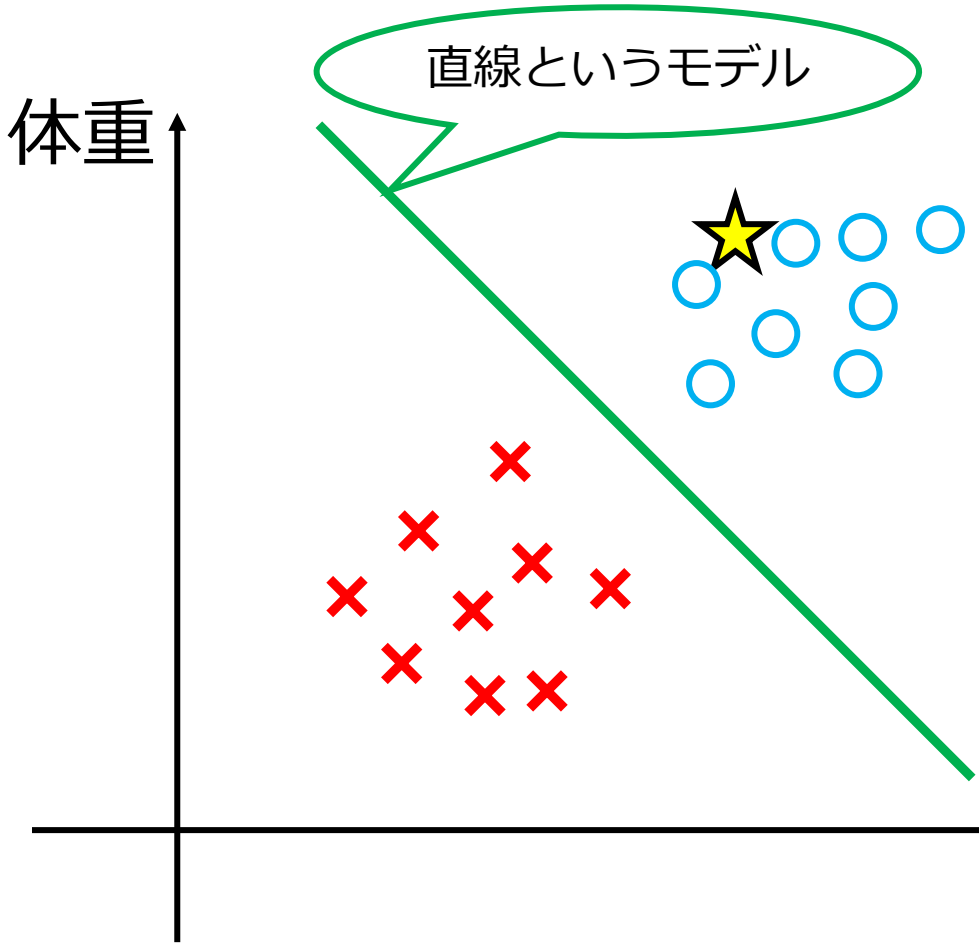
○ 小学６年生

× 小学１年生

☆ 何年生？

人間はいろんな方法で簡単にとける！
機械には手順を教えてあげないといけない

身長と体重から学年を当てる問題



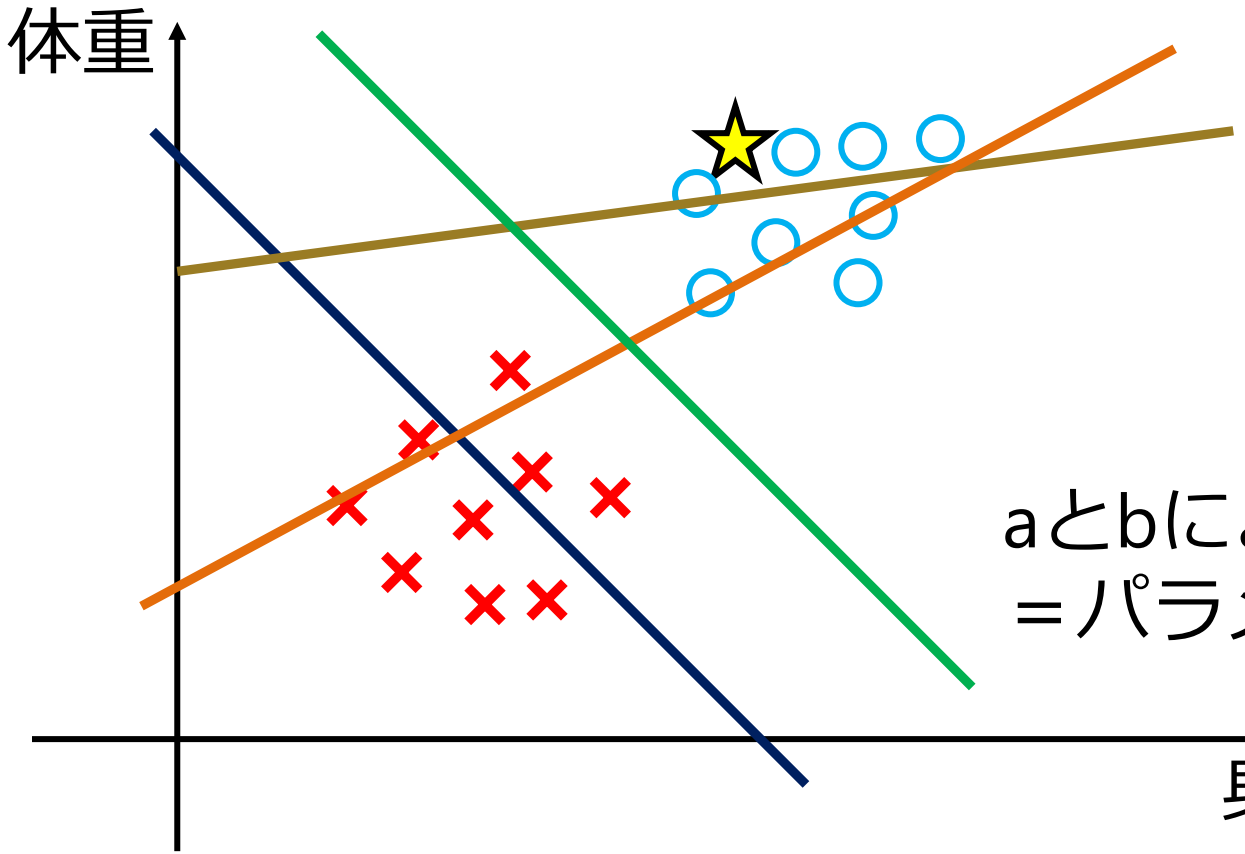
○ 小学６年生

× 小学１年生

☆ 何年生？

境界線の上か・下にあるかで判断できる
では，どのように境界線を決める？

身長と体重から学年を当てる問題



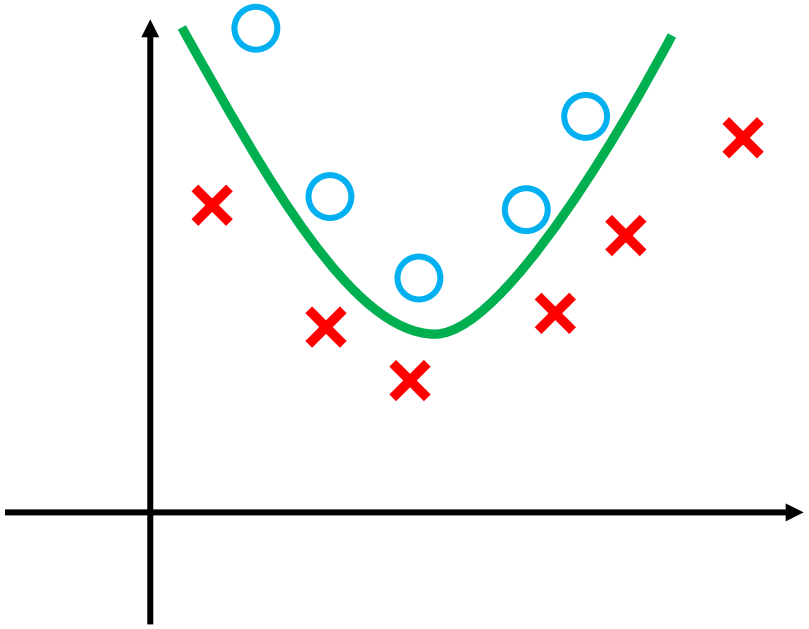
元のデータを綺麗に
分離できる境界線を探す

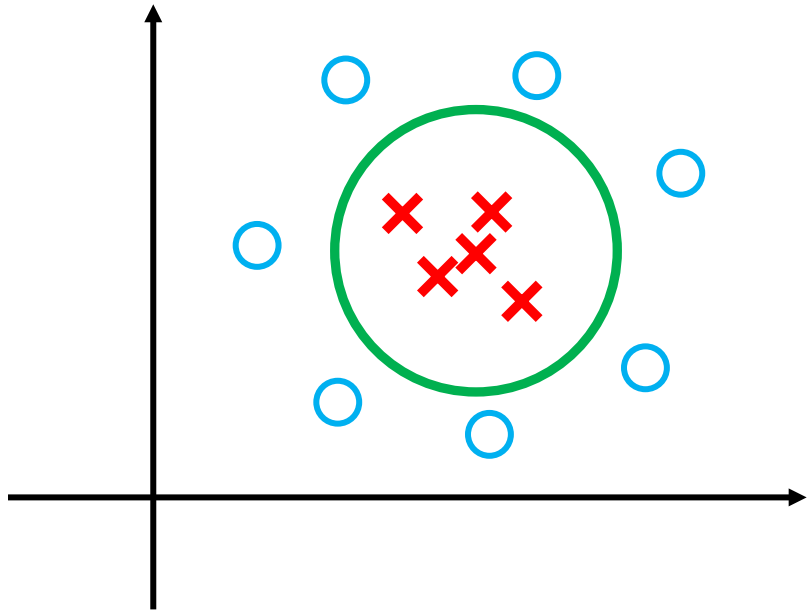
体重 = a×身長+b

aとbによって直線の形が変わる
＝パラメータaとbを調整する

境界線のモデル

- 直線では対応できないケースが一般的
  - 現実のデータの特徴分布の形はよくわからないことがほとんど



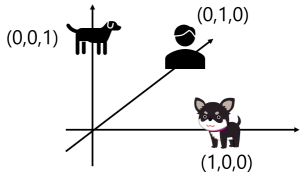


特徴設計

- 特徴設計がよければ，直線でも分離できるが．．．
  - 何を特徴にするのか？
  - もっぱら特徴はベクトルとして記述されるが，その次元数は？

One-hot エンコーディング

- ある要素だけが1で，それ以外は0のベクトル表現
  - ベクトルの次元数は，単語数とする
  - 10000語で構成される言語では10000次元ベクトル

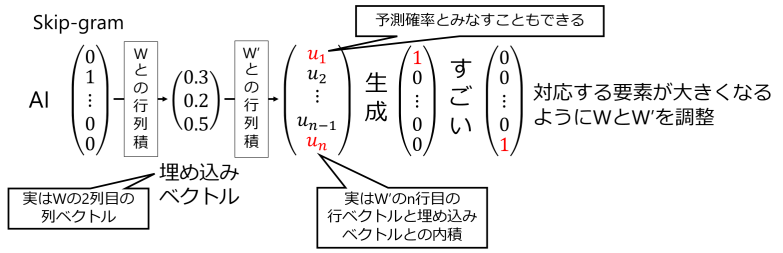


犬・男・チワワの3単語
どれも直線距離は同じ

犬とチワワの意味の近さ
を考慮できない

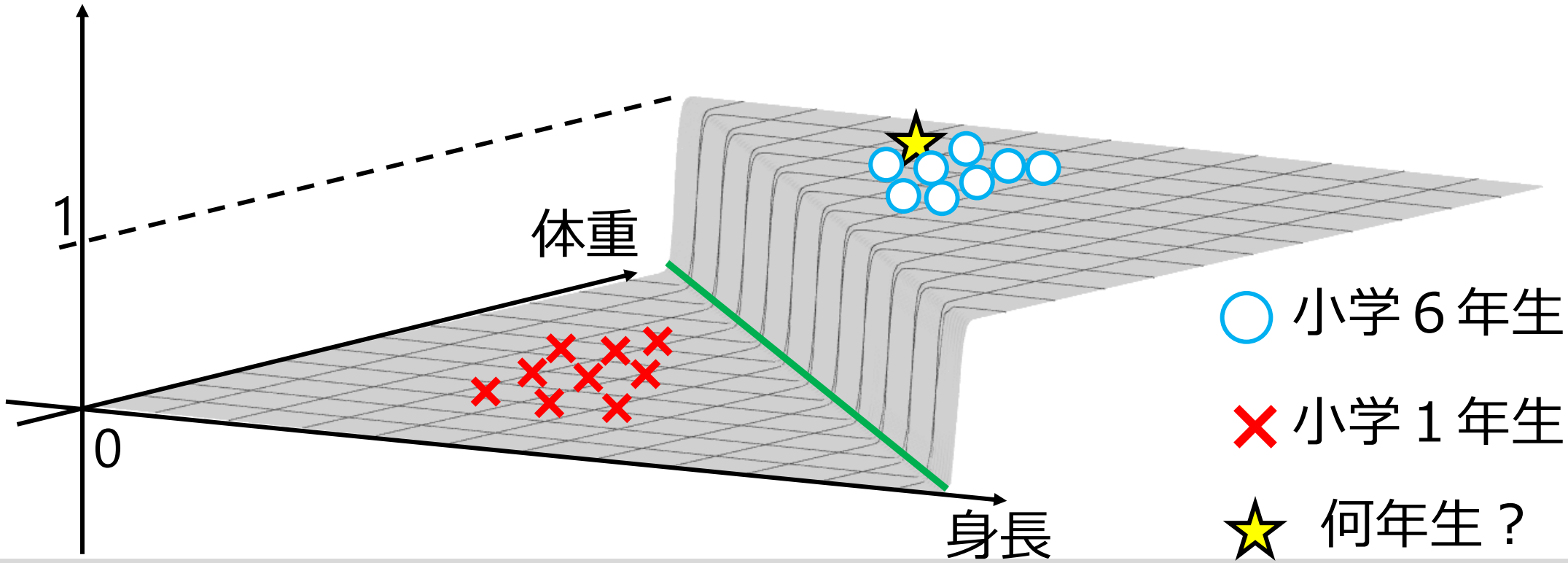
Word2Vec

- 単語の関係性を持ったベクトルを学習する
  - 埋め込み表現（Embedding）・分散表現



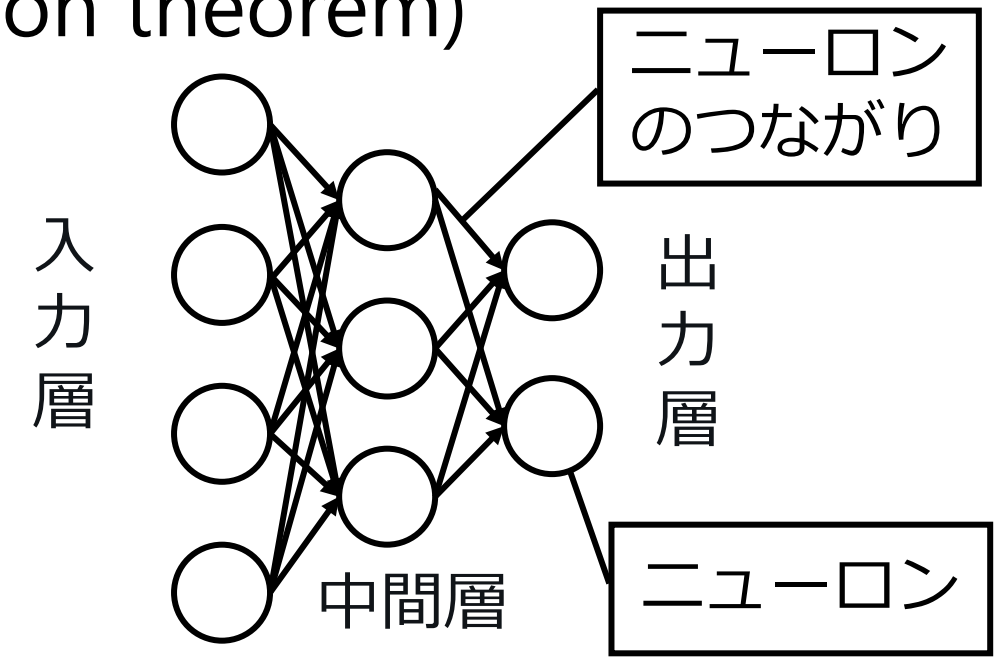
別視点：身長と体重から学年を当てる問題

- 6年生なら1，1年生なら0を出力する関数でも分類できる
  - 関数 f(身長，体重)が図のようになるようにパラメータを調整する
  - 「この入力には，この出力をせよ」



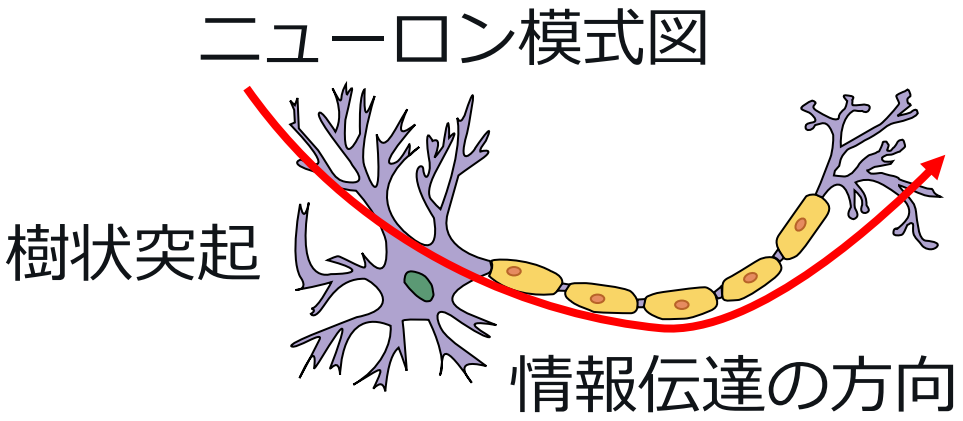
ニューラルネットワーク

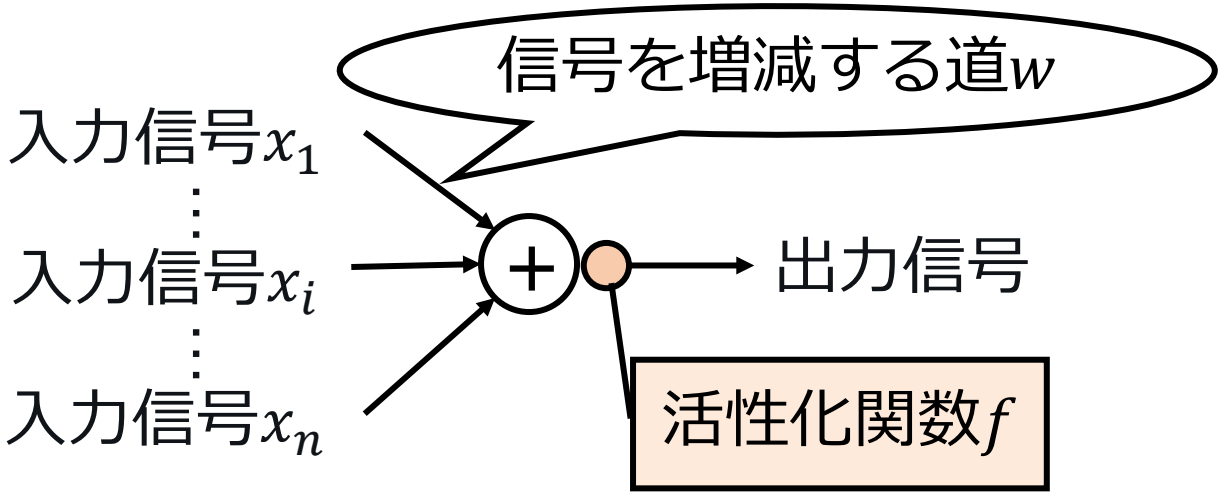
- ニューロンを参考に考案されたモデル
  - 入力層・中間層・出力層から構築される
  - 各層のニューロン数・中間層の数は任意に設定可能
- 普遍性定理（Universal approximation theorem)
  - 少なくとも一つの中間層があり，
    その中間層が十分なニューロンを持てば，
    任意の連続関数を近似可能



ニューロン

- 神経細胞をモデル化
  - 一定以上の強さの信号を受け取ると発火し，次のニューロンに情報が伝わる
  - 入力信号を増幅したり，減衰したりして，発火をコントロール



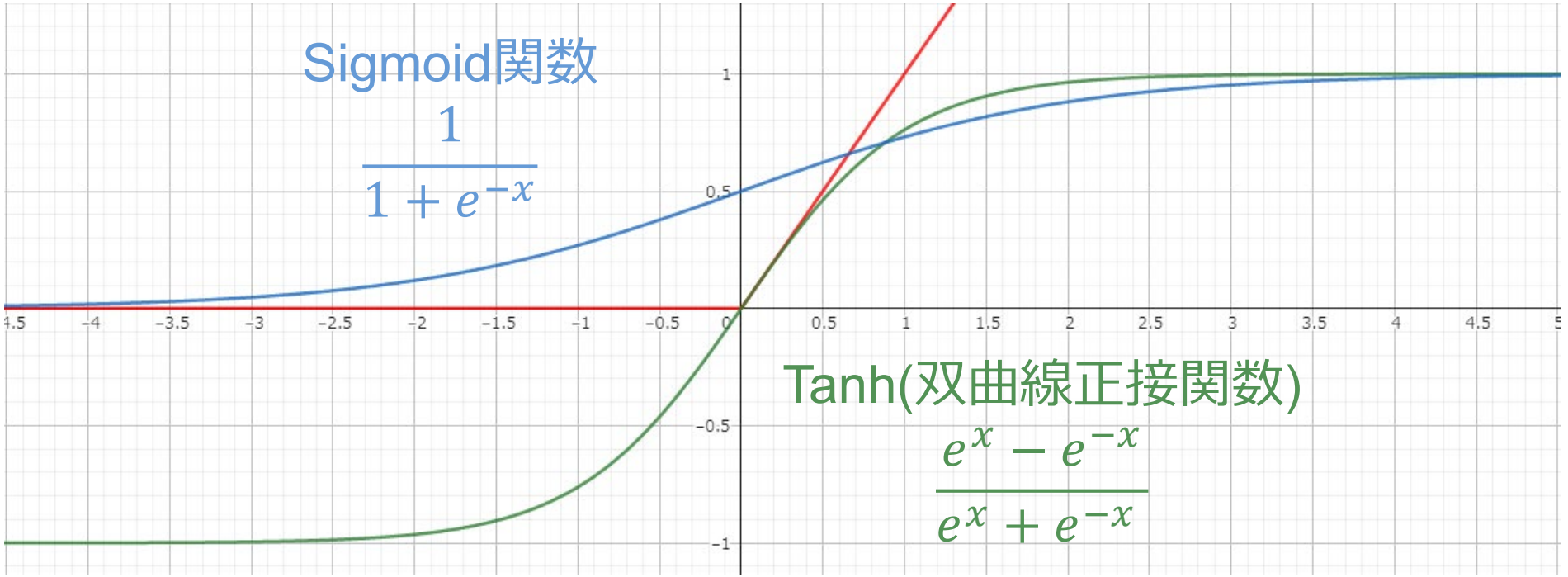


$$f(w_1x_1 + \cdots + w_ix_i + \cdots + w_nx_n)$$

活性化関数

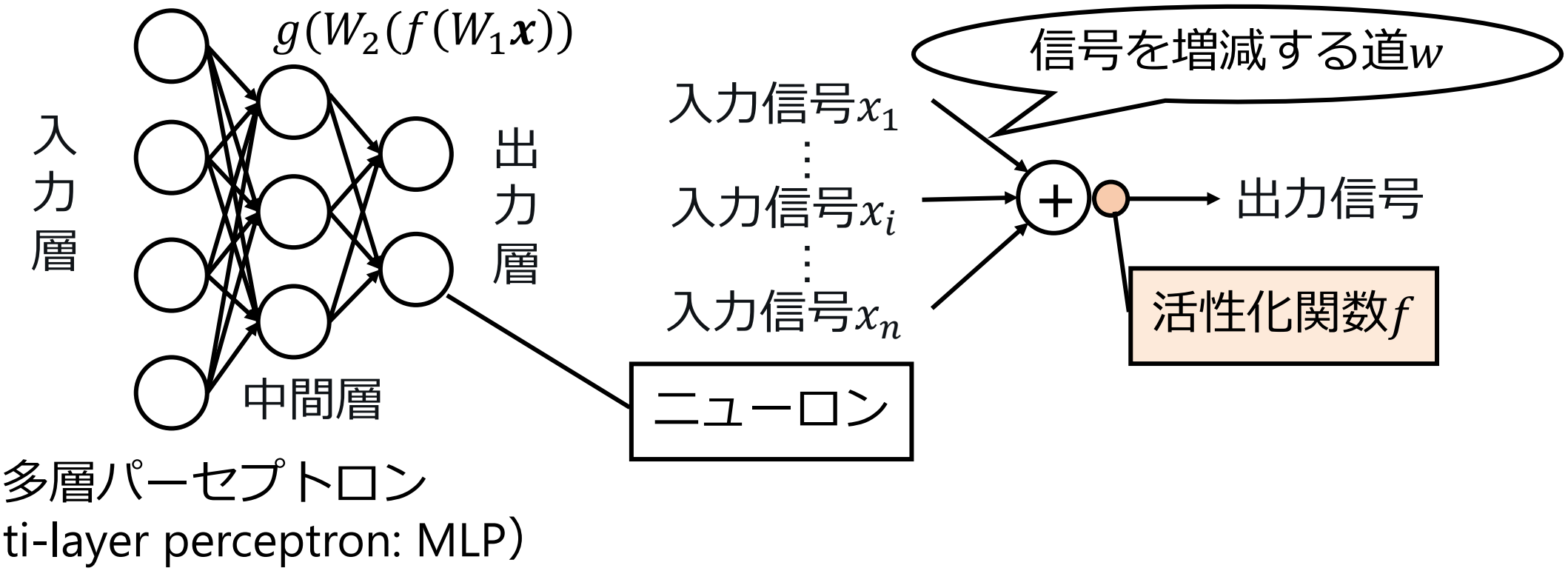
- （劣）微分可能な非線形関数
  - 発火のための閾値処置を担う

ReLU関数
$\max(0, x)$



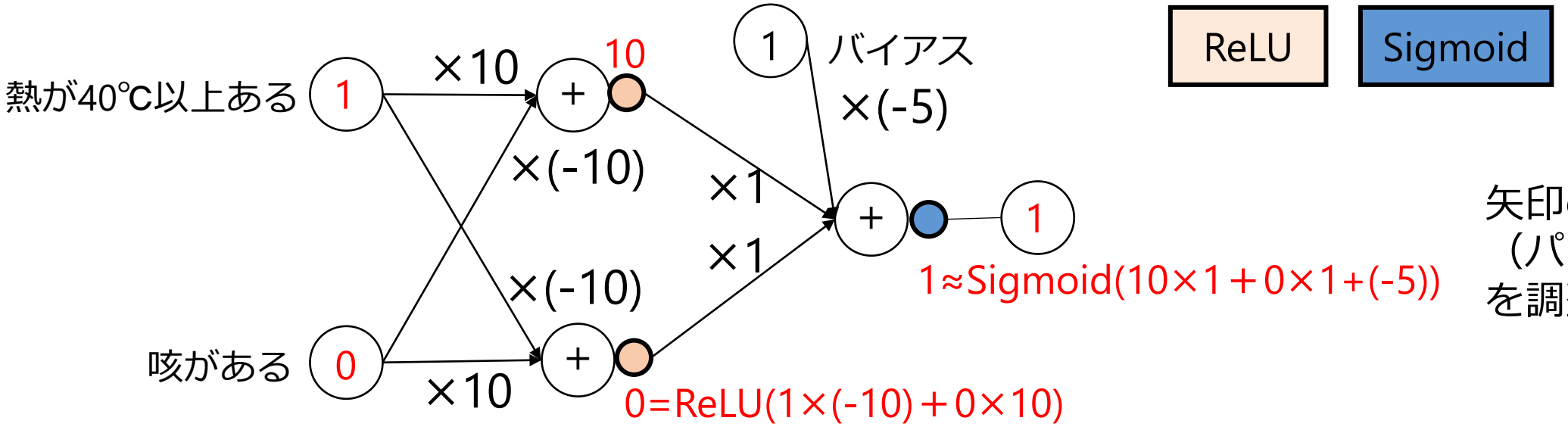
ニューラルネットワークの結局

- 増減した入力信号の和に非線形関数を適用することを順番に沢山・何度も繰り返すだけ



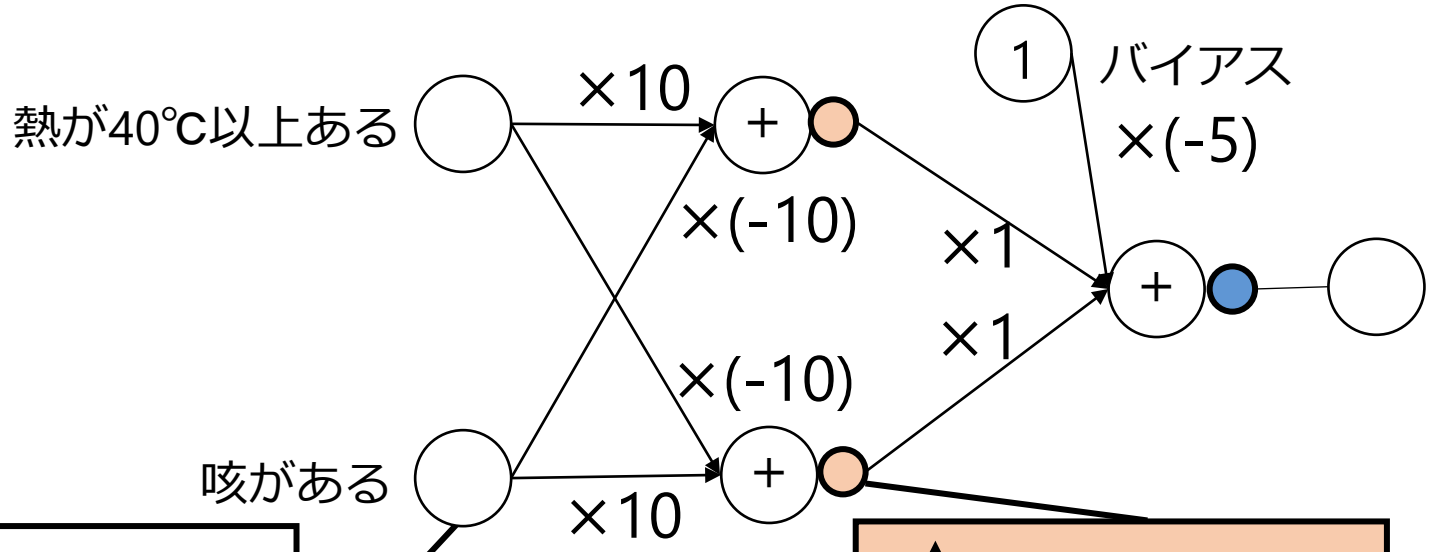
ニューラルネットワークの推論過程

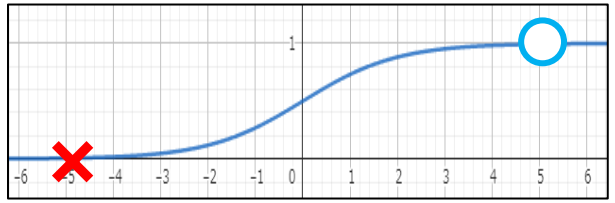
①熱が40℃以上ある	②咳がある	インフルエンザ以外の病気チェック
YES＝ 1	YES＝1	NO＝0
YES＝ 1	NO=0	YES＝1
NO=0	YES＝ 1	YES＝1
NO=0	NO=0	NO=0

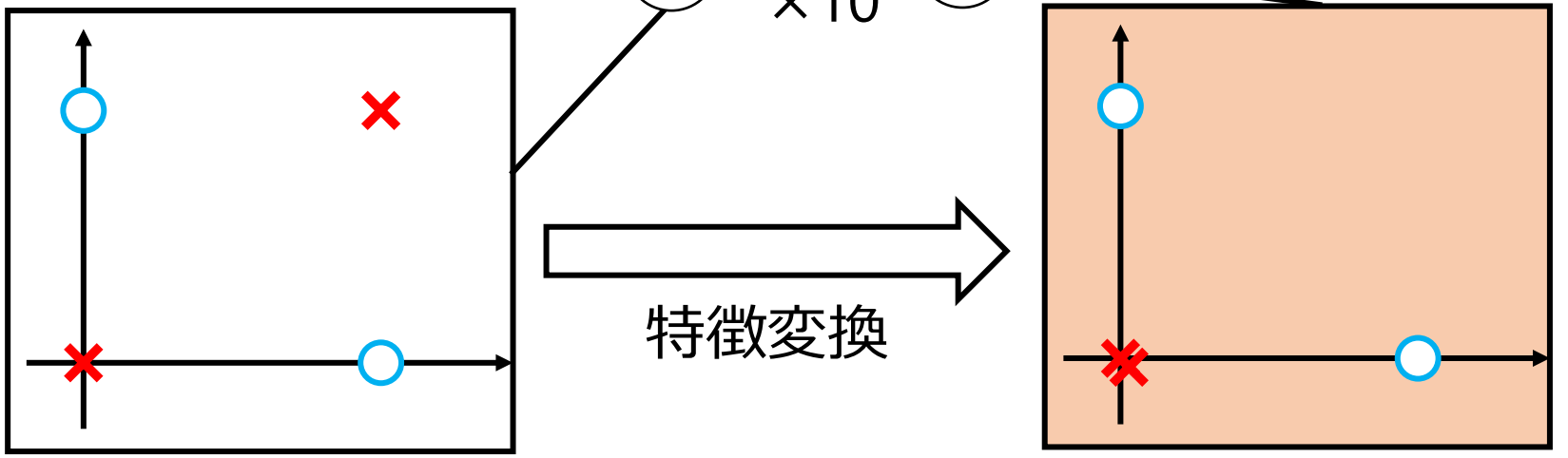


中間層の特徴

①熱が40℃以上ある	②咳がある	インフルエンザ以外の病気チェック
YES＝1	YES＝1	NO＝0
YES＝1	NO=0	YES＝1
NO=0	YES＝1	YES＝1
NO=0	NO=0	NO=0







入力層での特徴空間

特徴変換

中間層での特徴空間

直線で分けることができるようになった！

特徴抽出→推定の同時最適化

- 普遍性定理と多層化による特徴変換
  - モデルに直線や2次関数などの選択を気にせず
    しかも，目的を達成できる特徴を抽出できる．．．

- パラメータ数の増大
  - 今やGPTなどビリオンクラスのパラメータ
  - 目的を達成するパラメータを調整できるようになるまでに
    様々なテクニックが生まれた
    - 正則化，初期化，バッチサイズ，学習率

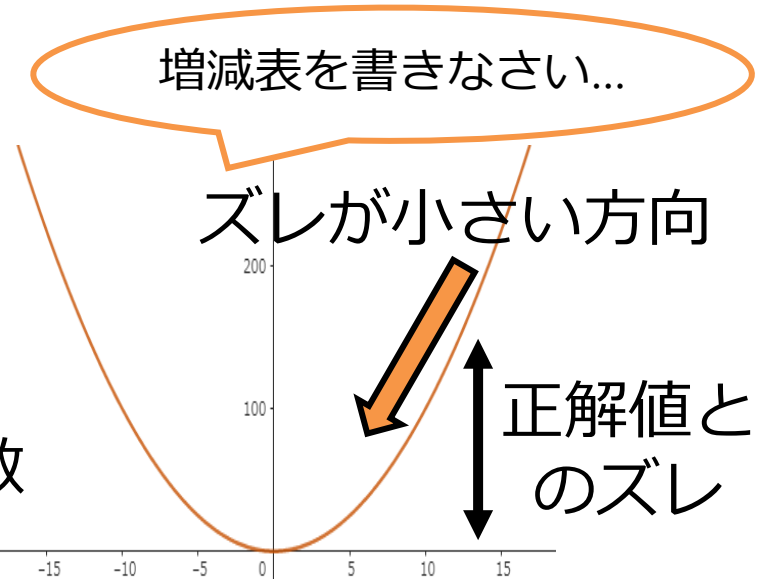
パラメータ調整のアイディア

- 目的の値に近づくようにパラメータを調整する
  - 例：(1,0)を入れたら，1が出力されるように
  - 最小二乗法
  - 勾配法・誤差逆伝播法

$$\sum_{\text{訓練データに対して}} (\text{正解値} - NN\text{の出力値})^2$$

NNも関数
合成関数の微分....

⇒ NNの出力値の関数



Transformer (Attention Is All You Need)

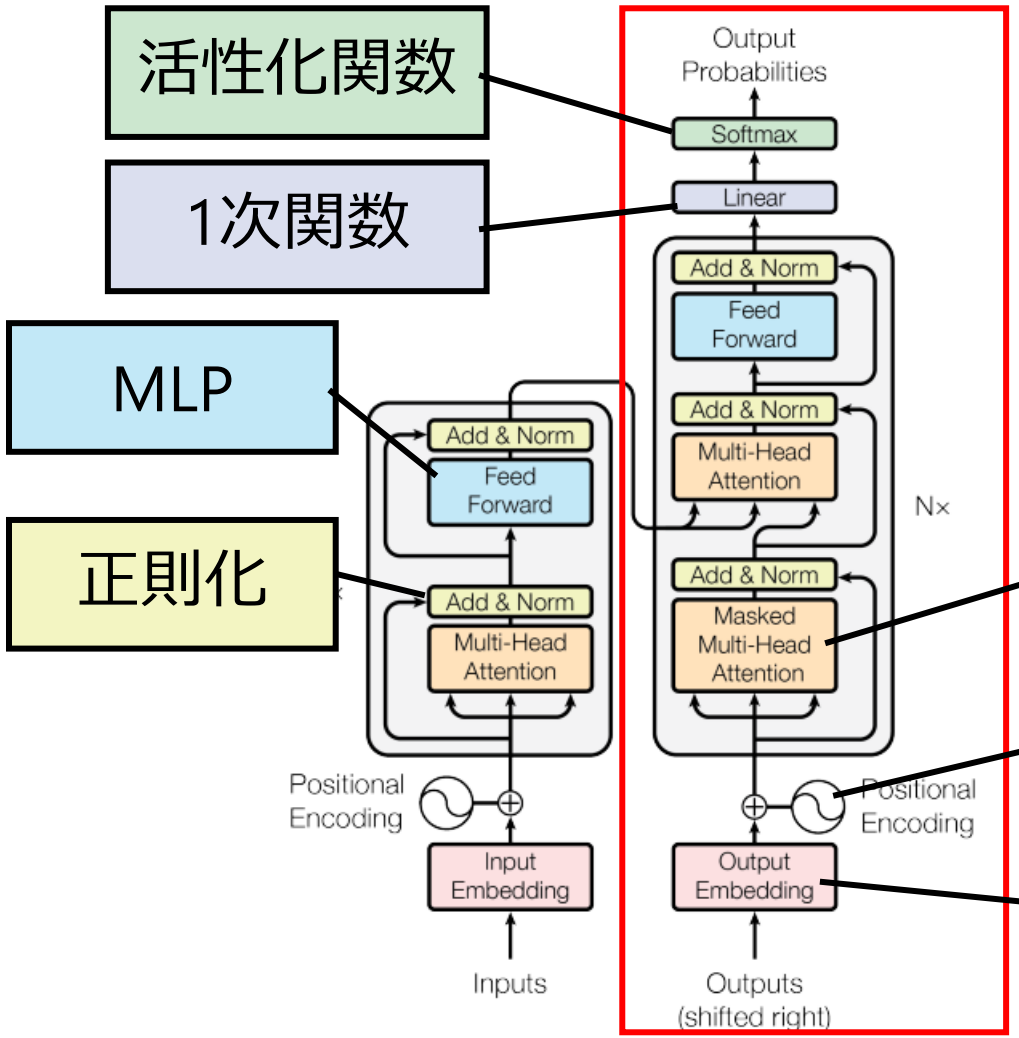


Figure 1: The Transformer - model architecture.

活性化関数

1次関数

MLP

正則化

注意機構

位置埋め込み

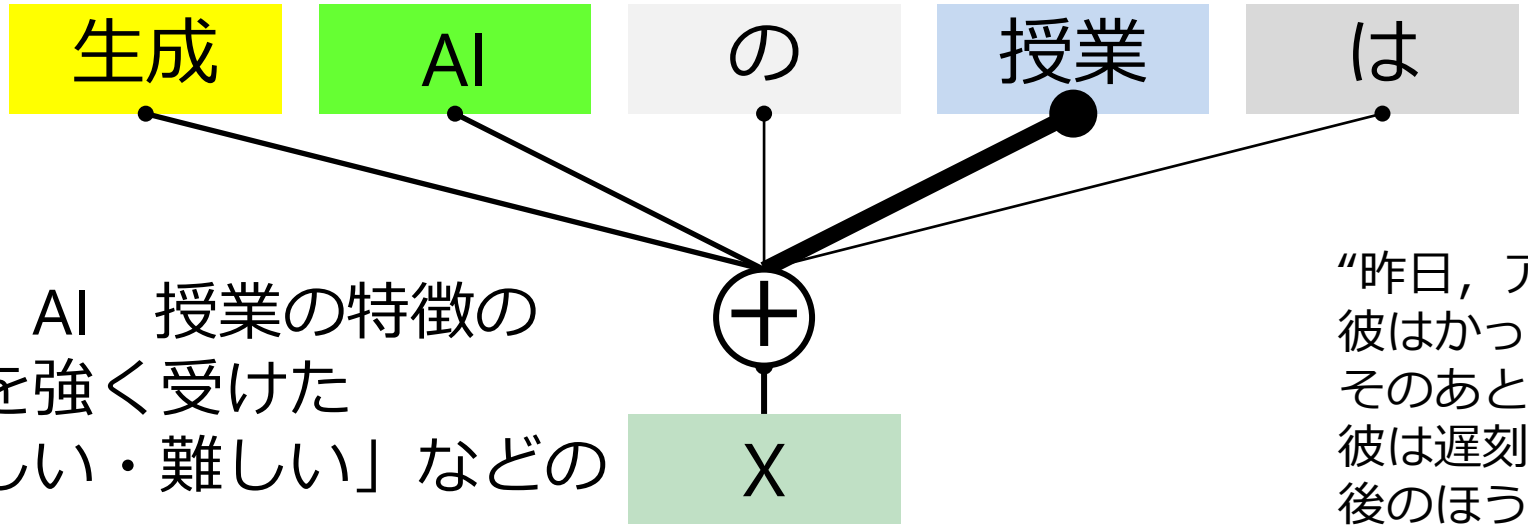
単語埋め込み

- 文章のような長い系列を効率的に扱うために登場
- GPTは右側だけ使う

これが主役！
文章中の単語の特徴を
単語の並び・組み合わせ
から順々に特徴変換

再掲：GPTの実態

- 前に現れた単語の埋め込み特徴の組み合わせで
  次の単語の特徴を計算
  - TransformerのAttention機構
  - 予測する単語以外の特徴も同様の仕組みで計算される



生成　AI　授業の特徴の
影響を強く受けた
「楽しい・難しい」などの

“昨日，アイドルに会いました.
彼はかっこよかった
そのあと，友達と待ち合わせをしていたけど,
彼は遅刻してきた.“
後のほうの「彼」の特徴の組み合わせは？

注意（Attention）

- 特徴を計算するとき，どの特徴に注意をむけるかを決める
  - 検索に似ている＝調べたいキーワード（クエリ）を入力して，
    データベースにあるキーワード（キー）を調べて，
    該当したキーワードの情報（バリュー）を出力する

クエリ：
ヨークシャーテリア

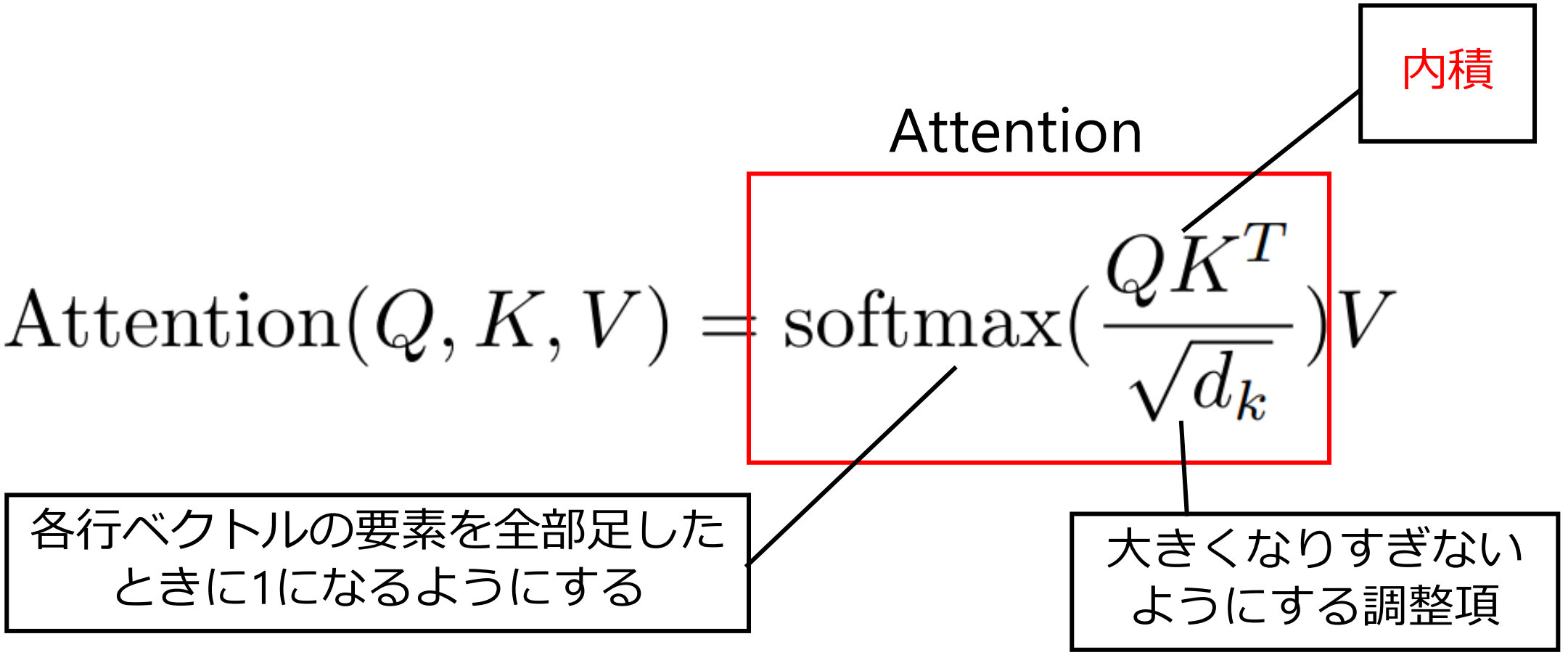
キーとクエリで
似てるものを検索

(50, 5, 10, 10)
該当したバリューを出力

キー	バリュー			
	体長	性格	毛の長さ	体重
チワワ	50	3	2	1
柴犬	100	45	2	10
サル	100	55	1	30
テリア	50	5	10	10
ヒト	170	10	1	70

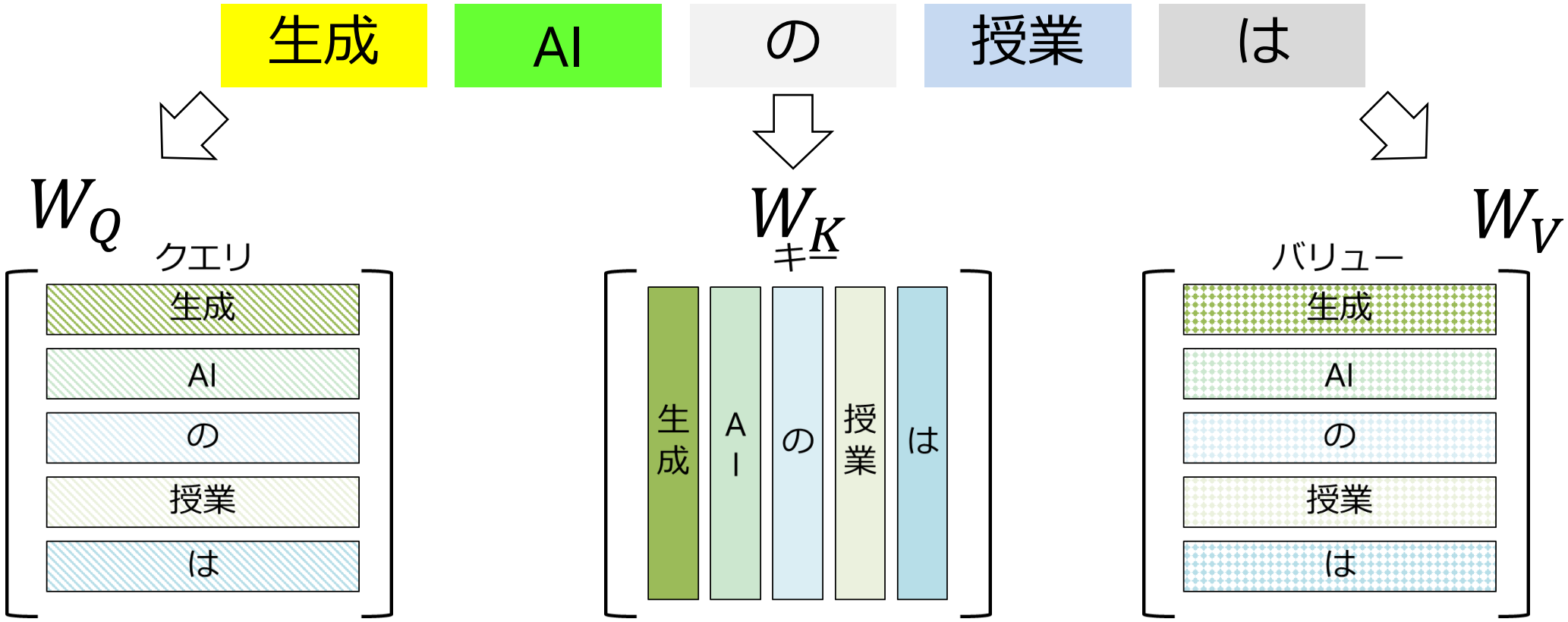
Scaled Dot-Product Attention

- Q（クエリ）・K（キー）・V（バリュー）は行列



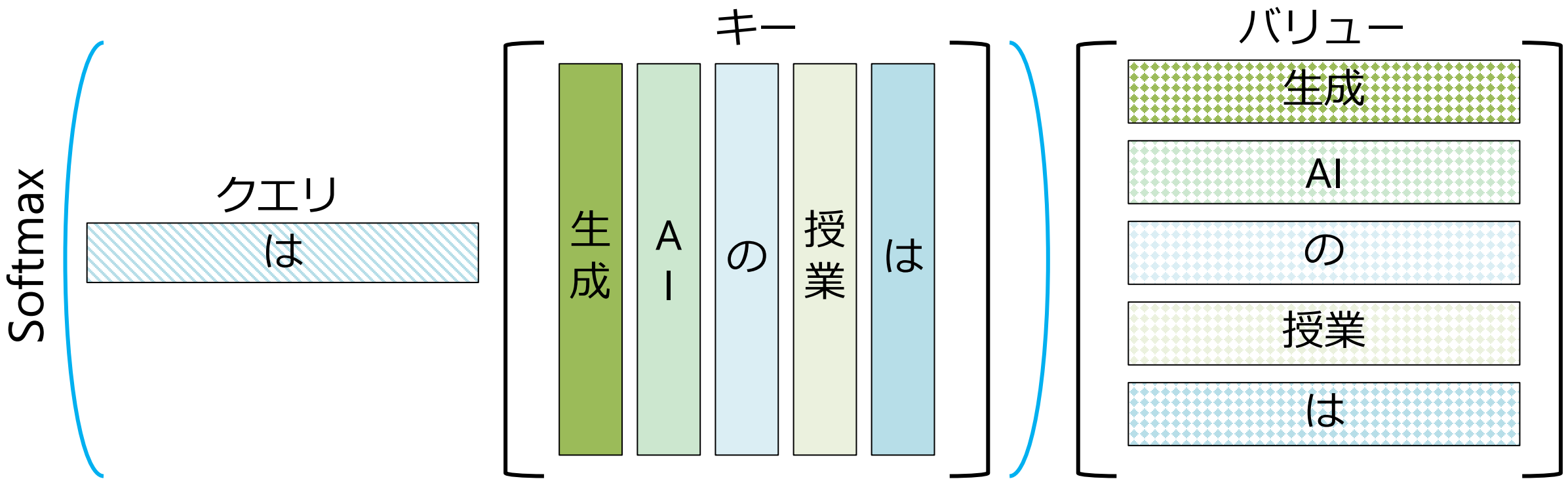
ＱＫＶの出どころ

- 各単語埋め込みベクトルをＱＫＶ用に線形変換
  - 線形変換：それぞれのベクトルとQKV用の行列との行列積



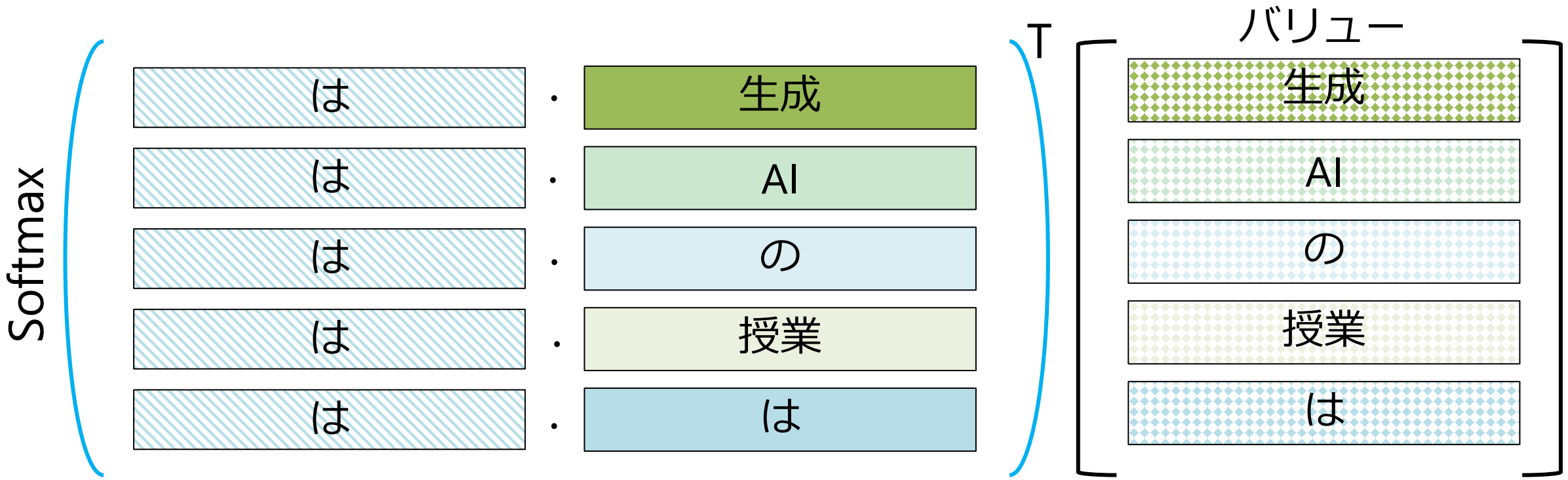
Scaled Dot-Product Attentionの流れ

- 簡単のため，Qの1行だけを考える
  - それぞれの棒は特徴ベクトル，dは省略



Scaled Dot-Product Attentionの流れ

- Softmaxの中身は特徴ベクトルの内積



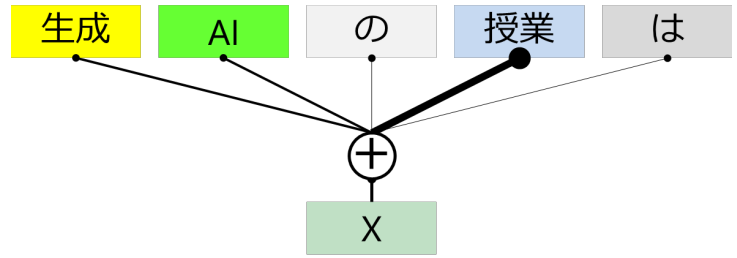
Scaled Dot-Product Attentionの流れ

- Softmaxは要素の大小関係を維持したまま
  要素を全部足したときに1になるようにする

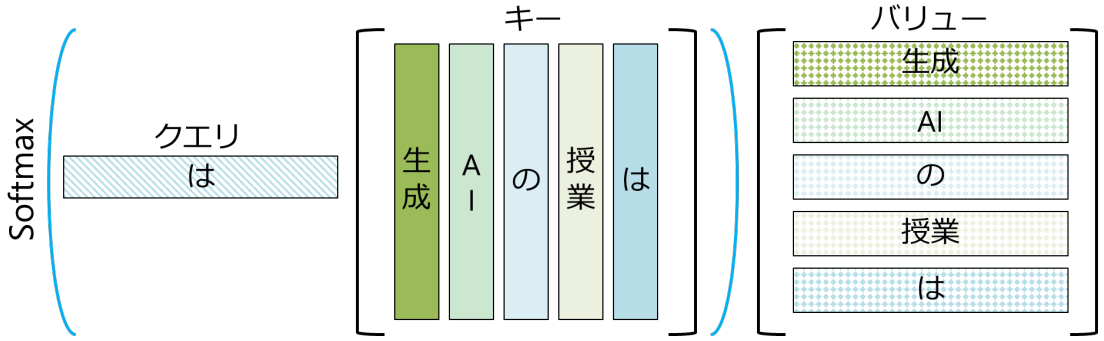
例：“授業“が多めに注目されて特徴が混ざる

Attentionの結局

- 単語埋め込みベクトルの類似度をもとに
  埋め込みベクトルの組み合わせを決定
  - 今回のAttentionをSelf-attentionという



- 「は」から新たに計算される特徴ベクトル
  は文章全体の特徴を反映していると考えられる
  - 次単語予測に用いられる



再掲：GPT-1を軽く読む

- Improving Language Understanding by Generative Pre-Training

3.1 Unsupervised pre-training

Given an unsupervised corpus of tokens $\mathcal{U} = \{u_1, \ldots, u_n\}$, we use a standard language modeling objective to maximize the following likelihood:

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta) \quad (1)$$

今は単語と思ってよし

調整パラメータ

自然な文章では
この値は大きくなる

GPT

予測する単語

予測する単語より前
に現れる単語列

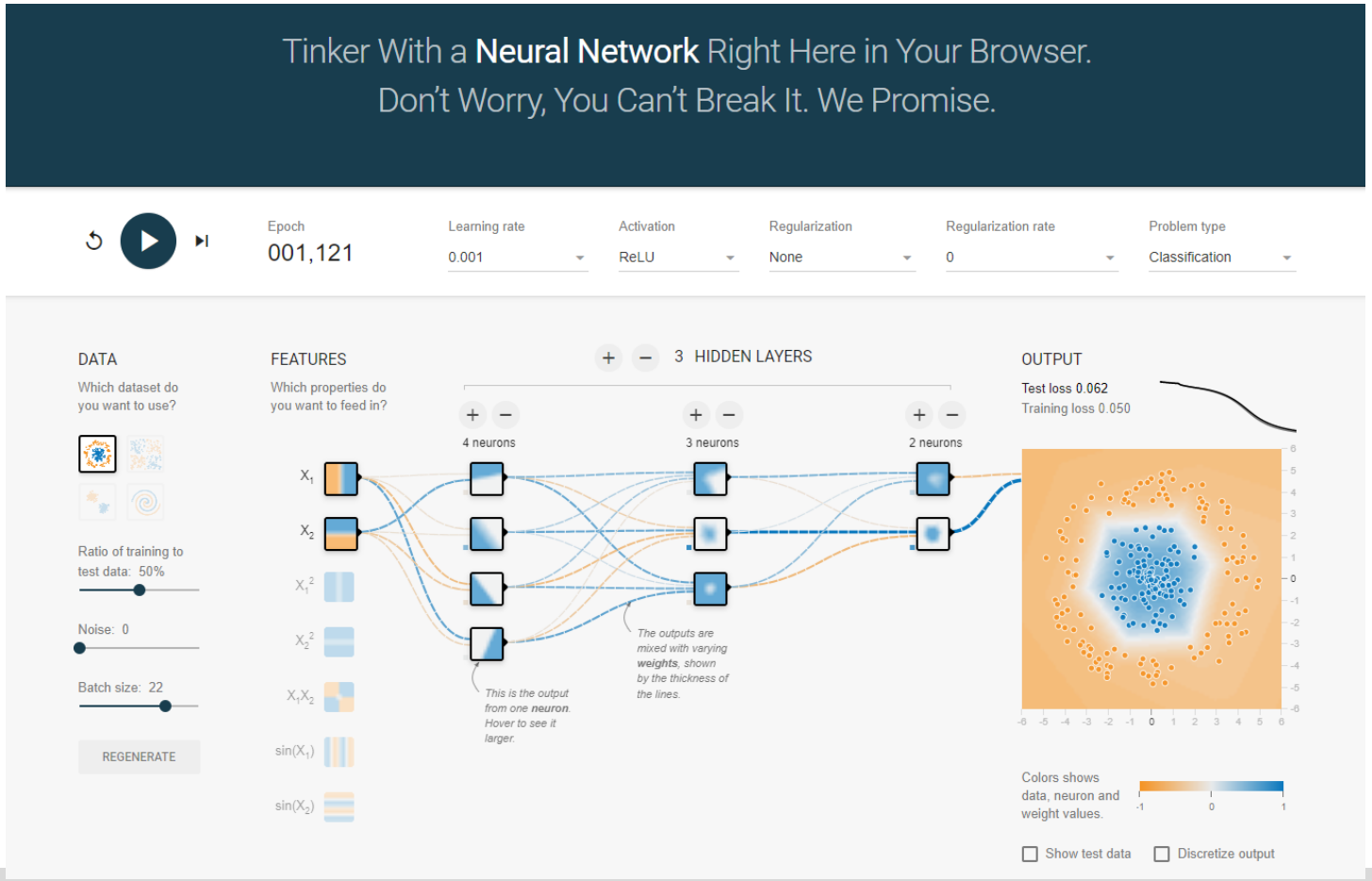
自然な文章の例：「生成AIの授業は楽しい」「生成AIの授業は難しい」
- GPT(犬｜生成)よりもGPT(AI｜生成)のほうがあり得る
- GPT(楽しい｜生成AIの授業は)とGPT(難しい｜生成AIの授業は)はどちらもあり得る

関連話題

- パターン認識
  - 「入力」が何かを判定する問題全般（ex. 犬の画像を犬と判定する）
- ニューラルネットワークの種類
  - 畳み込みNN（CNN），リカレントNN（RNN）
- 誤差逆伝播法
  - DNNのパラメータの勾配を求める方法
- 学習を成功させるための各種テクニック
  - L2正則化・バッチ正則化・初期化・Dropout・最適化手法（Adam）・残差接続・データ拡張など
- 機械学習の形式
  - 教師あり機械学習・教師なし機械学習・強化学習

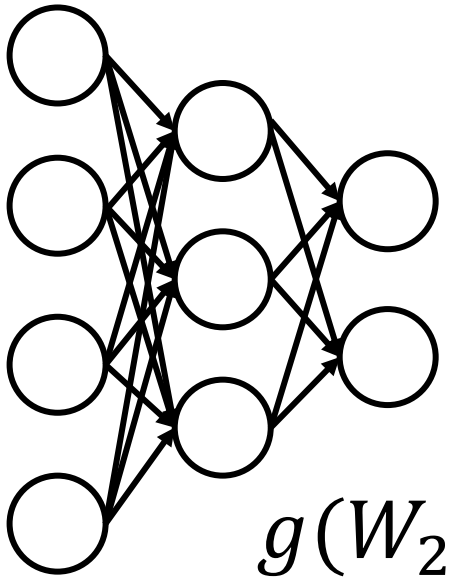
Google playground

- https://playground.tensorflow.org/



余談：AttentionとMLPの関係

- W_1 と W_{att}の違いはなんであろうか？



$g(W_2(f(W_1\boldsymbol{x}))$

$$W_{att}\,\mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$