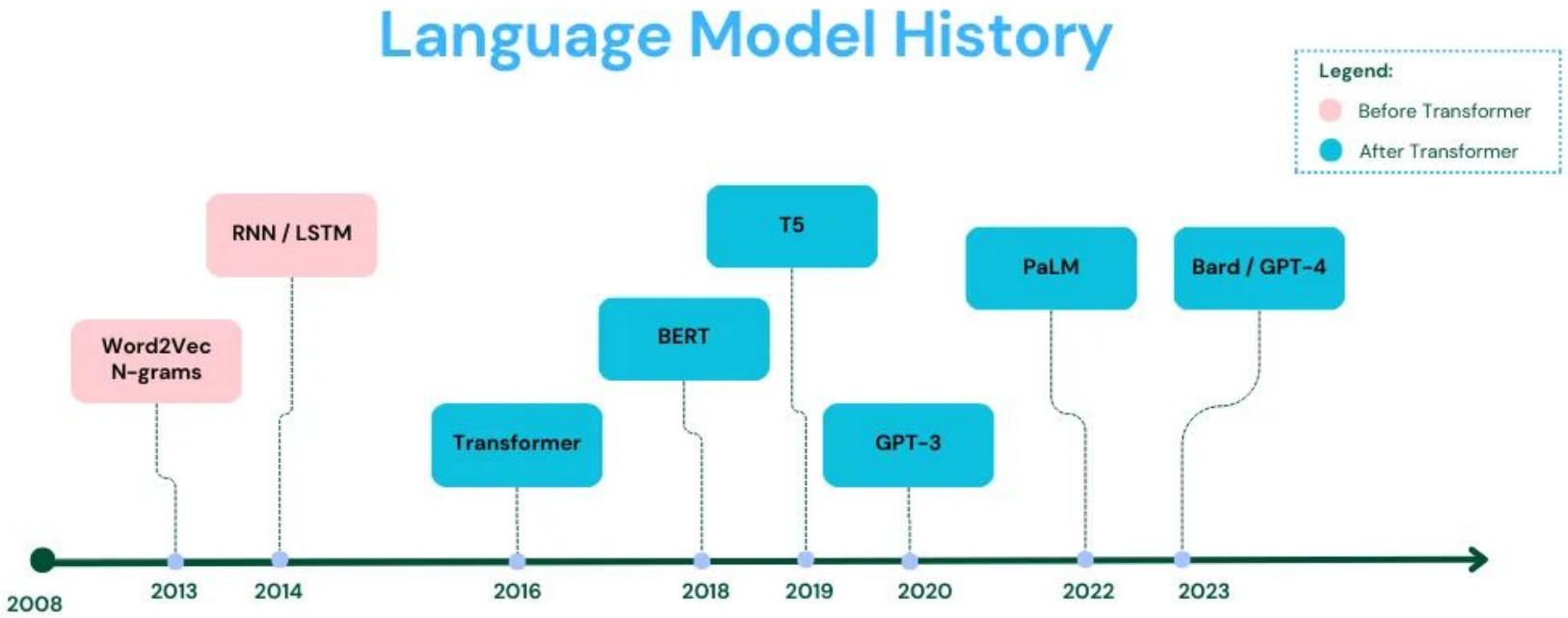
テキスト生成AIモデル
LSTMによる文章生成の仕組み

言語モデルの歴史


Language Model History
Legend:
Before Transformer
After Transformer
Word2Vec
N-grams
RNN / LSTM
Transformer
BERT
T5
GPT-3
PaLM
Bard / GPT-4
2008
2013
2014
2016
2018
2019
2020
2022
2023

テキスト生成AIモデルの目標

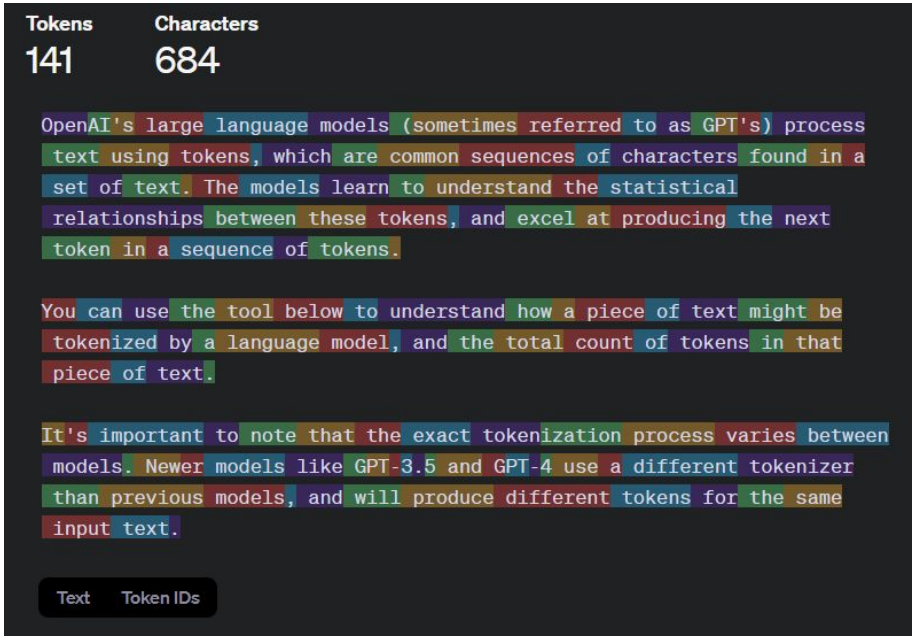
入力

- <START> I like an apple. You like an

コンテキストウインドウ
（モデルが一度に処理できるトークン数）

出力

- orange . <END>



続きの文字列を予測したい

使用した訓練データ

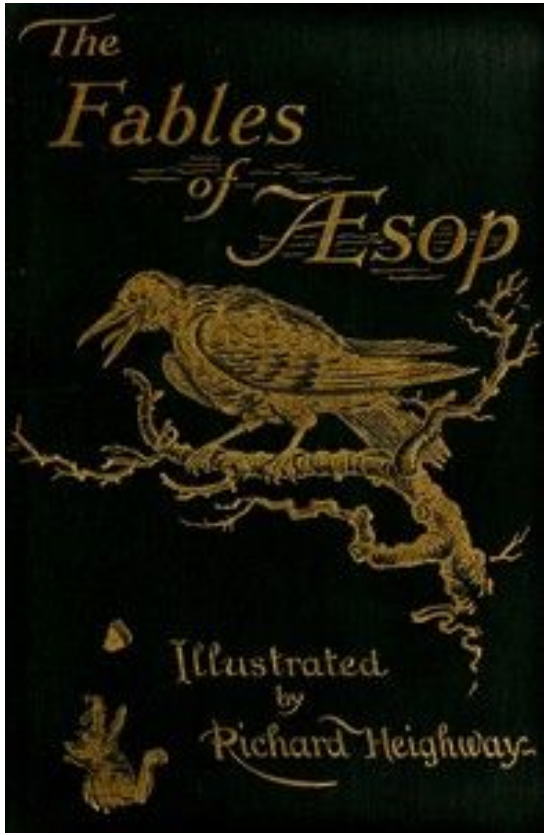
イソップ物語の英文（単語数46万、語彙数4169）

https://www.gutenberg.org/ebooks/28

PREFACE

It is difficult to say what are and what are not the Fables of Æsop. Almost all the fables that have appeared in the Western world have been sheltered at one time or another under the shadow of that name. I could at any rate enumerate at least seven hundred which have appeared in English in various books entitled _Æsop's Fables_. L'Estrange's collection alone contains over five hundred. In the struggle for existence among all these a certain number stand out as being the most effective and the most familiar. I have attempted to bring most of these into the following pages.

There is no fixed text even for the nucleus collection contained in this book. Æsop himself is so shadowy a figure that we might almost be forgiven if we held, with regard to him, the heresy of Mistress Elizabeth Prig. What we call his fables can in most cases be traced back to the fables of other people, notably of Phædrus and Babrius. It is usual to regard the Greek Prose Collections, passing under the name of Æsop, as having greater claims to the eponymous title; but modern research has shown that these are but medieval prosings of Babrius's verse. I have therefore felt at liberty to retell the fables in such a way as would interest children, and have adopted from the various versions that which seemed most suitable in each case, telling the fable anew in my own way.



前処理：テキストの標準化

一例

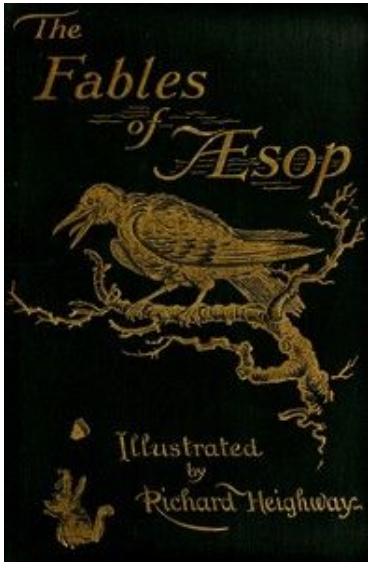
- 全文字を小文字に変換
- 章の区切り文字 “\n\n\n\n\n” を
  シードテキスト “||||||||||||||||||” に置換
- 改行文字 “\n” を1つの空白文字 “ “ に置換
- 1つ以上続く空白文字を “. ” に置換
- 2つ以上続くピリオド “..” を “.” に置換
- “,”や”.”など記号を “[記号][空白文字]“に置換
- 2つ以上続く空白文字を1つに置換

変換後の一例

|||||||||||||||||| the fox and the grapes . one hot summer ’ s day a fox was strolling through an orchard till he came to a bunch of grapes just ripening on a vine which had been trained over a lofty branch . “ just the thing to quench my thirst , ” quoth he . |||||||||||||||||| the peacock and juno . a peacock once placed a petition before juno desiring to have the voice of a nightingale in addition to his other attractions ; but juno refused his request .

前処理：トークン化

単語の出現回数をカウントして単語とインデックスの対応辞書を作成



対応辞書(4169単語)

‘|’ : 1,
‘,’ : 2,
‘the’ : 3,
‘and’ : 4,
‘.’ : 5,
‘a’ : 6,
‘to’ : 7,
‘“” : 8,
‘of’ : 9,
‘he’ : 10,

対応辞書を使って
テキストをトークン化

1, |
3, the
56, fox
4, and
3, the
940, grapes
5, .
6, a
382, hungry

前処理：訓練で使う入力データ作成

シードテキスト

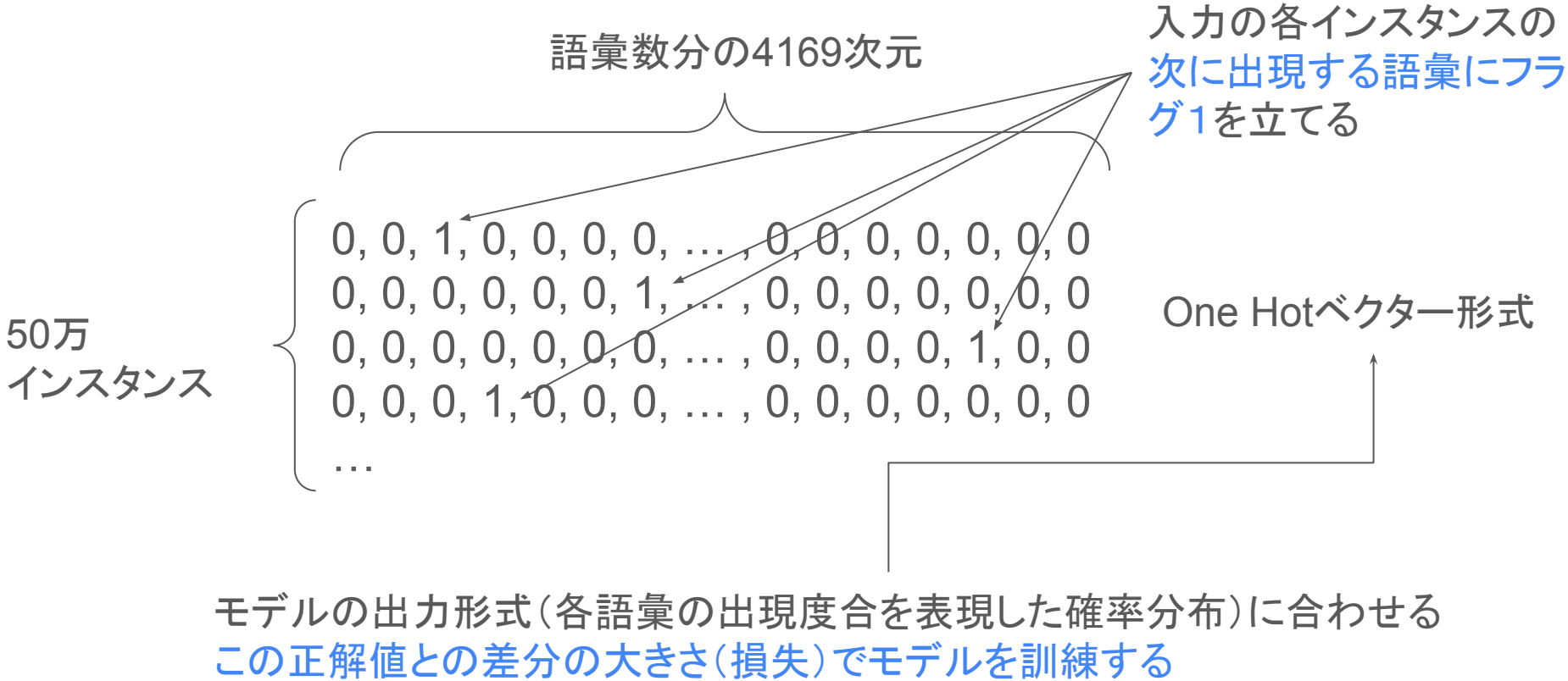
the fox and the grapes . one hot summer ’ s day a fox was strolling through an orchard till

50万インスタンス

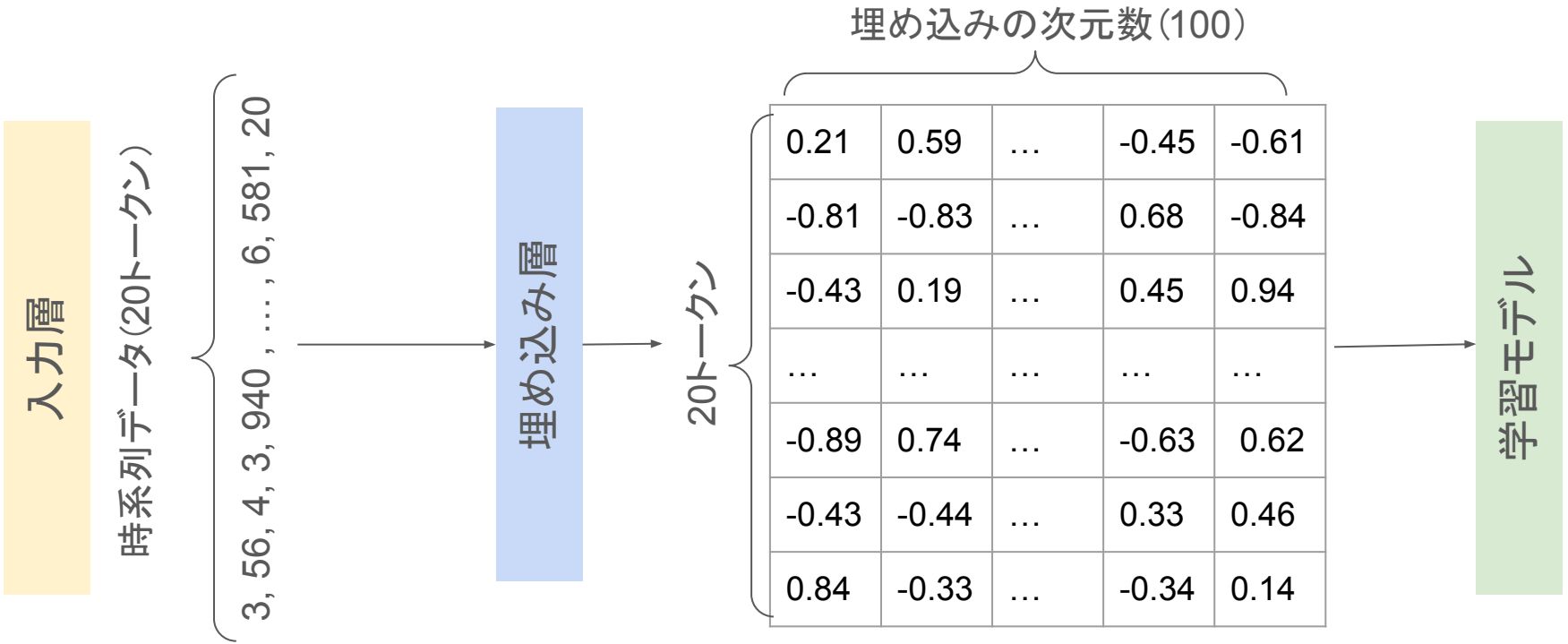
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
3, 56, 4, 3, 940, 5, 6, 382, 56, 94, 77, 216, 1557, 9, 940, 941, 62, 6, 581, 20,
12, 2226, 162, 6, 359, 2227, 2, 4, 158, 11, 250, 7, 383, 35, 29, 1176, 25, 359, 25, 10,
88, 55, 3, 582, 5, 19, 16, 12, 37, 14, 785, 2, 17, 23, 47, 96, 43, 9, 383, 30,
28, 10, 170, 36, 425, 2, 4, 426, 89, 21, 57, 582, 9, 1558, 4, 2228, 2, 1559, 2, 8,
…

コンテキストウインドウ（今回は20トークン）

前処理：訓練で使う正解値ラベルの作成



埋め込み層を流れる時系列データ



なぜトークンから埋め込みを生成する必要があるのか

単語のインデックス番号の時系列トークンは離散的数値であり学習に使えない
埋め込みとは：整数トークン用のルックアップテーブル
学習中に埋め込みは更新されていく→結果、類似した語彙の値が近寄っていく

語彙数（4169）

トークン	埋め込み（次元数100）				
1	-0.13	0.45	…	0.13	-0.04
2	0.22	0.56	…	0.24	-0.63
…	…	…	…	…	…
4168	0.16	-0.70	…	-0.35	1.02
4169	-0.98	-0.45	…	-0.15	-0.52

OpenAI Embedding API

Embedding models

OpenAI offers two powerful third-generation embedding model (denoted by `-3` in the model ID). You can read the embedding v3 announcement blog post for more details.

Usage is priced per input token, below is an example of pricing pages of text per US dollar (assuming ~800 tokens per page):

MODEL	~ PAGES PER DOLLAR	PERFORMANCE ON MTEB EVAL	MAX INPUT
text-embedding-3-small	62,500	62.3%	8191
text-embedding-3-large	9,615	64.6%	8191
text-embedding-ada-002	12,500	61.0%	8191

← 1536次元

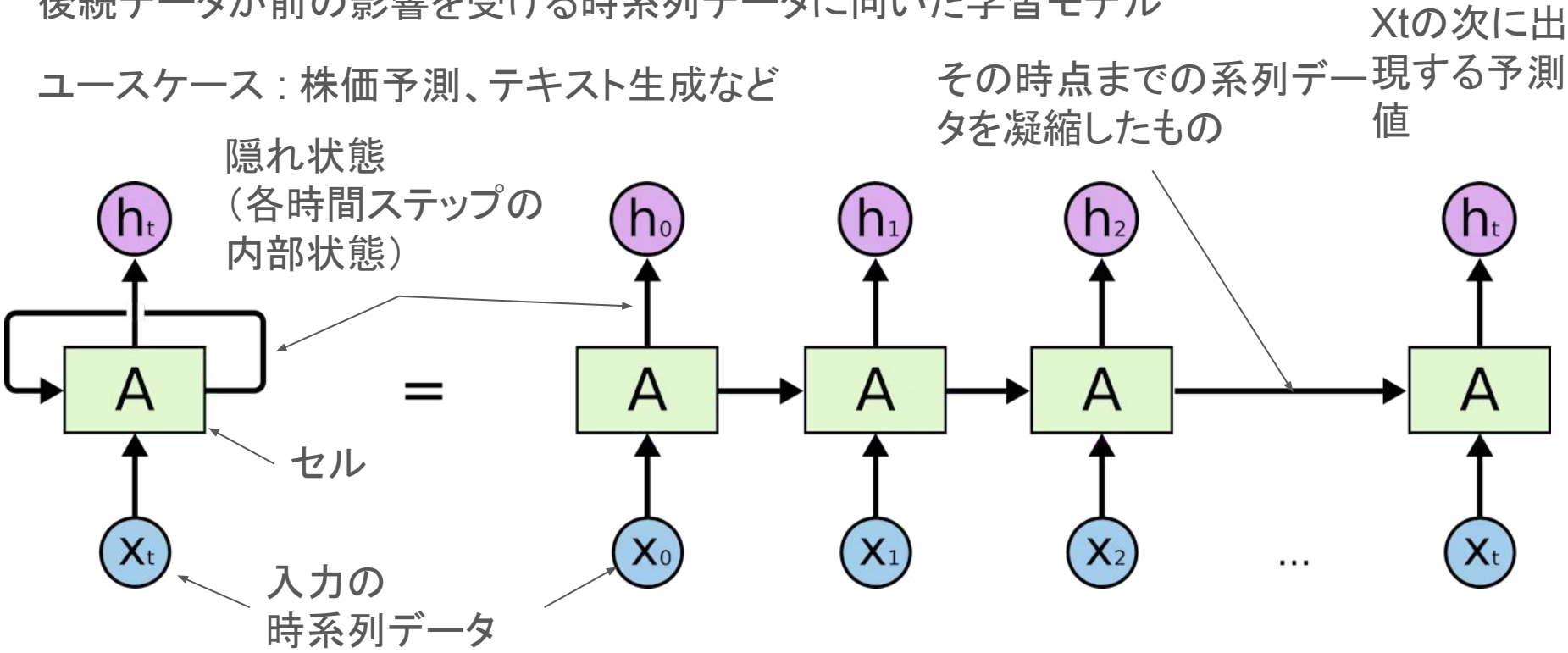
← 3072次元

← 現在使用不可

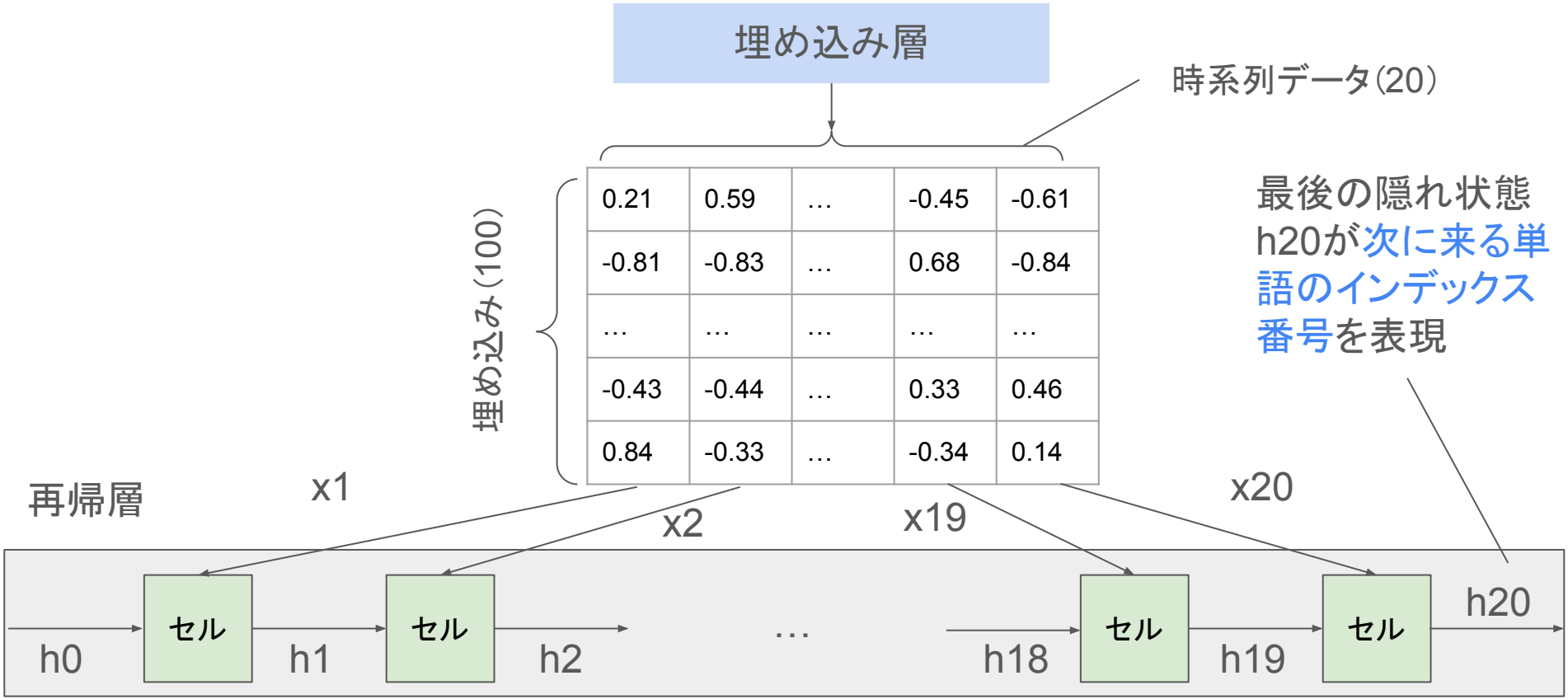
RNN リカレントニューラルネットワーク

後続データが前の影響を受ける時系列データに向いた学習モデル

ユースケース：株価予測、テキスト生成など



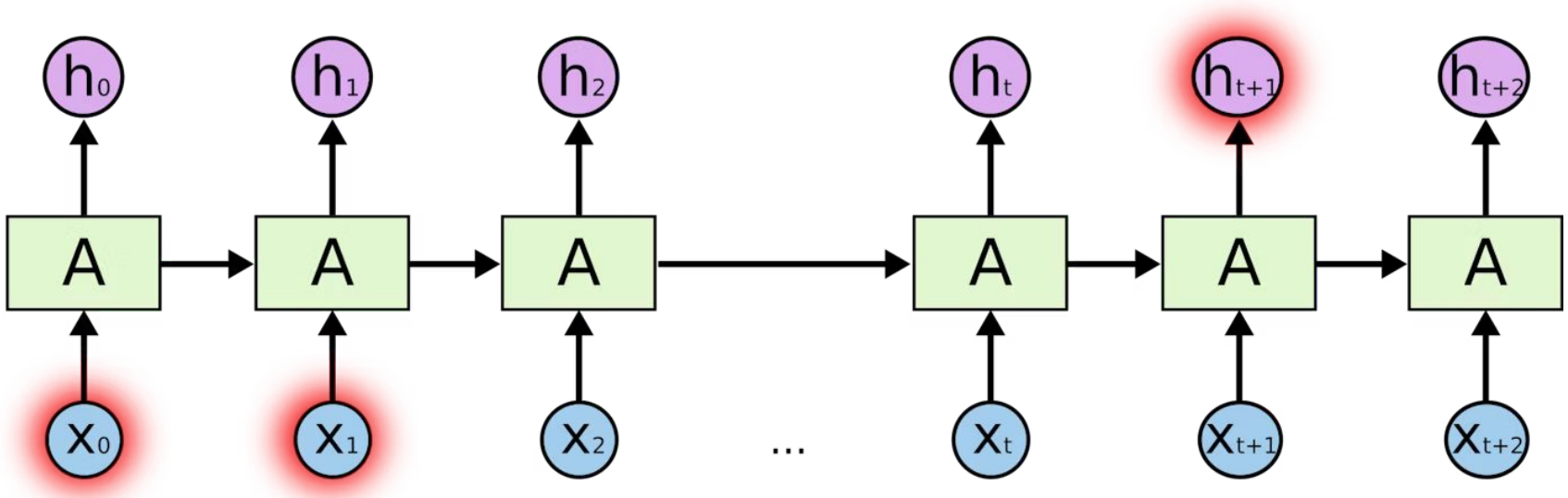
再帰層における1つの時系列データの流れ



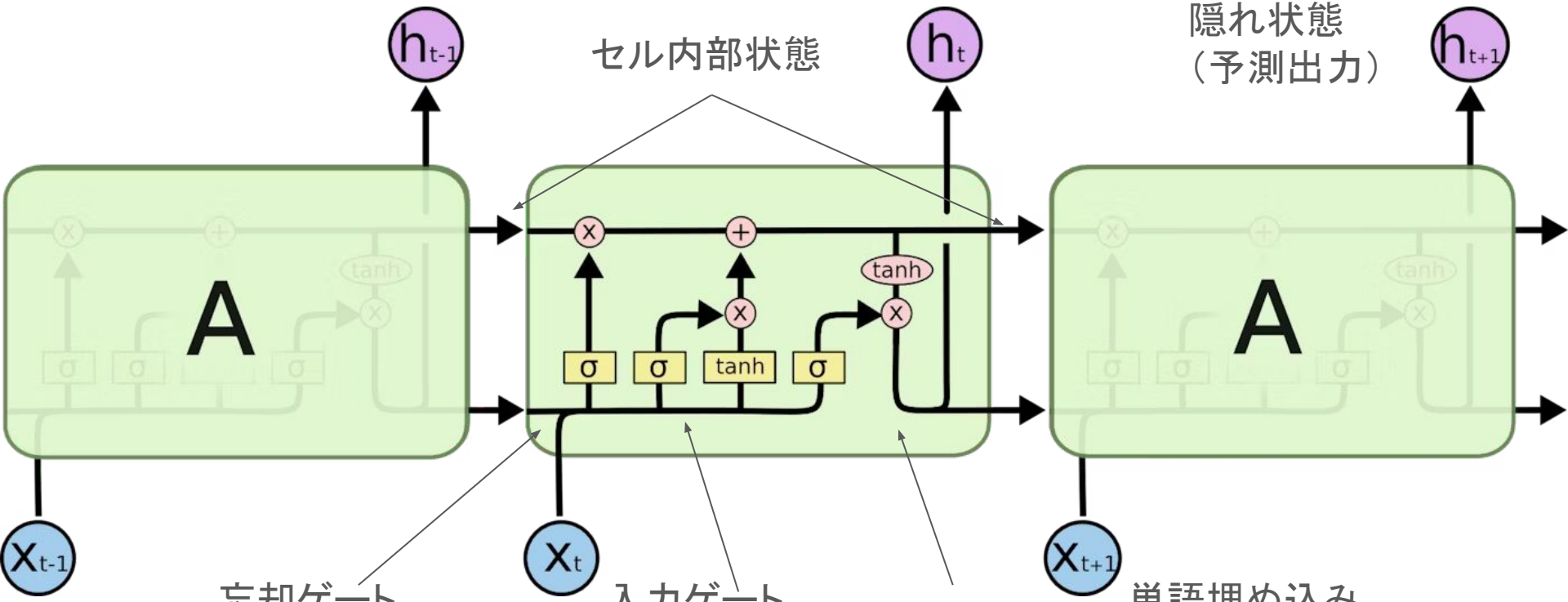
RNNの課題

データをセルに繰り返し入出力する過程で、流れる値の大きさがどんどん減衰

→ 時系列を大きく跨ぐ関係性を表現できなくなる（昔の記憶が薄れる）



LSTM : Long Short Term Memory（長・短期記憶）



LSTMを1層だけ使った学習モデルの一例

```
n_units = 256 # LSTMセル内のノード数
embedding_size = 100 # 埋め込みサイズ

text_in = Input(shape = (None,))
embedding = Embedding(total_words, embedding_size) # total_words x embedding_size のルックアップテーブル
x = embedding(text_in)
x = LSTM(n_units)(x)
# x = Dropout(0.2)(x)
text_out = Dense(total_words, activation = 'softmax')(x) # 語彙それぞれの次に来る確率

model = Model(text_in, text_out)

opti = RMSprop(learning_rate=0.001)
model.compile(loss='categorical_crossentropy', optimizer=opti)
```

重み総数

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, None)]            0

 embedding (Embedding)       (None, None, 100)         416900

 lstm (LSTM)                 (None, 256)               365568

 dense (Dense)               (None, 4169)              1071433

=================================================================
Total params: 1853901 (7.07 MB)
Trainable params: 1853901 (7.07 MB)
Non-trainable params: 0 (0.00 Byte)
```

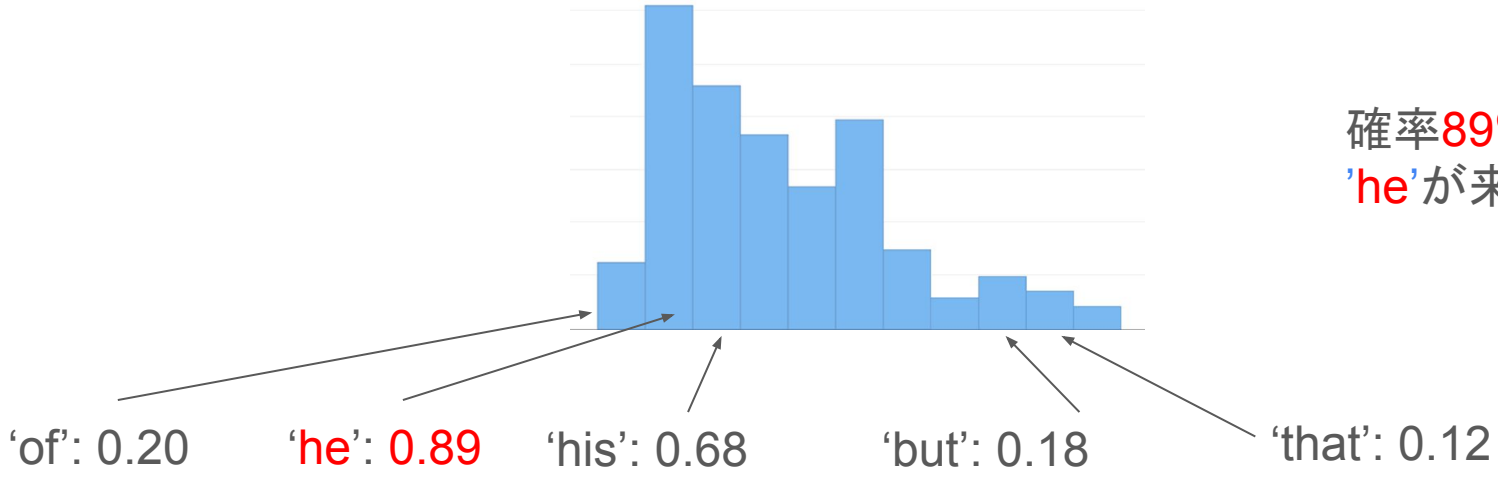
前処理、埋め込みの次元数、LSTM層の数、各LSTM内のニューロン数など調整箇所多数

学習済みモデルを用いた出現単語の予測

入力テキスト（単語数20）
- the fox and the grapes . one hot summer ’ s day a fox was strolli through an orchard till

入力テキストのトークン（トークン数20）
- 3, 56, 4, 3, 940, 5, 6, 382, 56, 94, 77, 216, 1557, 9, 940, 941, 62, 6, 581, 20

出力
- LSTMの最後の隠れ層からの出力
- 各語彙について次に出現する確率を示した確率分布（語彙数4169分の確率データ）



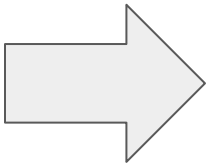
確率89%で次に
’he’が来ると分かる

次に来る単語の出現確率の一例（200エポック学習後）

入力：
the frog and the snake . the frog went to

次に来る単語トップ10
14.0% ：the
1.9% ：beast
1.5% ：attend
1.1% ：family
1.1% ：shepherd
1.1% ：fire
1.0% ：net
0.9% ：earth
0.8% ：partridge
0.7% ：apes

“the” を選択した場合



入力（ theを末尾に追加して左に1ずらす ）：
frog and the snake . the frog went to the

次に来る単語トップ10
17.2% ：back
11.7% ：horse
10.7% ：dogs
8.2% ：fox
6.4% ：make
4.1% ：ass
2.9% ：caught
2.9% ：take
2.0% ：head
1.7% ：both

物語の生成例

1. シードテキスト“||||||||||||||||||||”
   - 入力トークン 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
   - 出力
     - 3 “The”
2. 入力テキスト“|||||||||||||||||| the”
   - 入力トークン 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3
   - 出力
     - 56 “fox”
3. 入力テキスト“|||||||||||||||||| the fox”
   - 入力トークン 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 56
   - 出力
     - 4 “and”
4. 入力テキスト“|||||||||||||||||| the fox and”
   - 入力トークン 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 56, 4
   - 出力
     - 3 “the”

以降繰り返すことによって、徐々にテキストが生成されていく
the fox and the grapes . one hot summer ’ s day a fox was strolling through an orchard till he

1エポック学習後の物語の生成例

| | | | | | | | | | | | | | | | | | | | | | the fox and the fox . a fox . a fox , and the fox , and the fox , and the fox , and the fox , and the man , and the fox , and the fox , and the lion , and the lion of the lion

キツネとキツネ。キツネ。キツネとキツネとキツネとキツネとキツネとキツネと人間とキツネとキツネとライオンとライオンのライオン

10エポック学習後の物語の生成例

| | | | | | | | | | | | | | | | | | | | | | the wolf and the lion . a lion was a tree , and was a tree of the ass , who was a lion , and the ass had a lion and the ass which it was the lion . the wolf was a tree , and , he was

狼とライオン。ライオンは木であり、ロバの木であり、ロバはライオンであり、ロバはライオンであり、ロバはライオンでした。狼は木であり、彼は

100エポック学習後の物語の生成例

| | | | | | | | | | | | | | | | | | | | | | the fox and the snake . a snake , in crossing a river , was carried away by the current , but managed to wriggle on to a bundle of thorns which was floating by , and was thus carried at a great rate down - stream . a fox

キツネと蛇。蛇は川を渡っているときに流れに流されましたが、流れていたイバラの束になんとか這い上がり、こうして下流にすごい速さで流されました。キツネ

200エポック学習後の物語の生成例

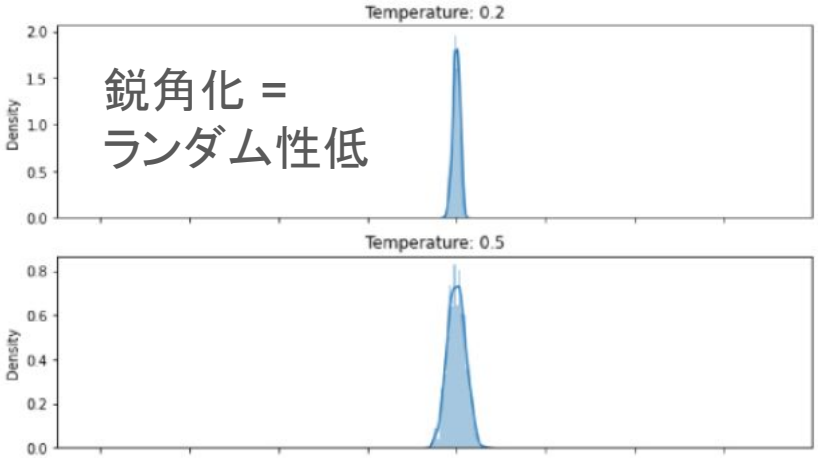
| | | | | | | | | | | | | | | | | | | | | | the ass and the dog . an ass and a dog were on their travels together , and , as they went along , they found a sealed packet lying on the ground . the ass picked it up , broke the seal , and found it contained some writing

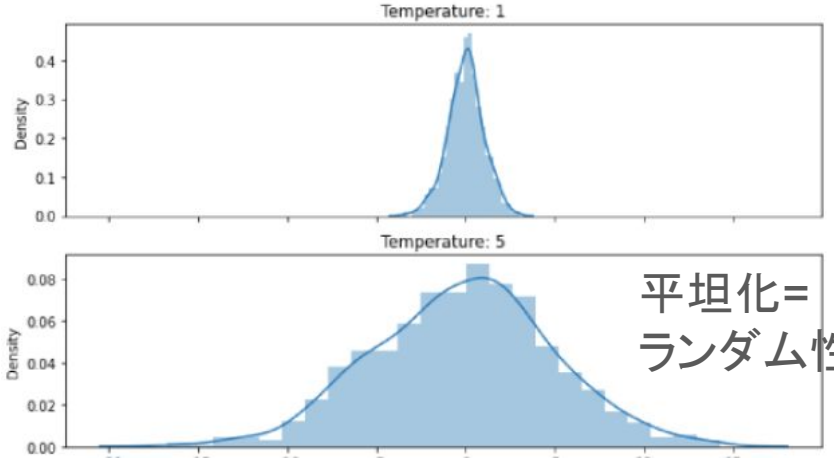
ロバと犬が一緒に旅をしていたのですが、歩いていると、地面に落ちている封印された包みを見つけました。ロバはそれを拾い上げて封印を破り、何か書いてあるのを見つけました。

temperatureによるランダム化の仕組み

Temperatureは各語彙の確率分布の形状を変化させる

- < 1 : 確率分布を鋭くし、もっと高い確率を持つ単語が選ばれやすくなる
- > 1 : 確率分布が平坦化し、ランダム性が増す
- 各語彙の確率を (1 / temperature) 乗してSoftmax





temperatureによる確率分布の変換例

preds : モデルの予測した確率分布

```
def sample_with_temp_v2(preds, temperature=1.0):
    # helper function to sample an index from a probability array
    preds = np.asarray(preds).astype('float64') # (4169,)
    preds = preds ** (1 / temperature)
    preds = preds / np.sum(preds)
    return np.random.choice(len(preds), p=preds)
```

temperatureを変化させた生成例（200エポック学習後）

0.1の場合

- the frog and the snake . the frog went to the back of the oak and seized her , and in order to be revenged with his treasure . as he lay gods , he was seen , who cried in despair , " alas ! of ! " it was creature , no friend , " said the and

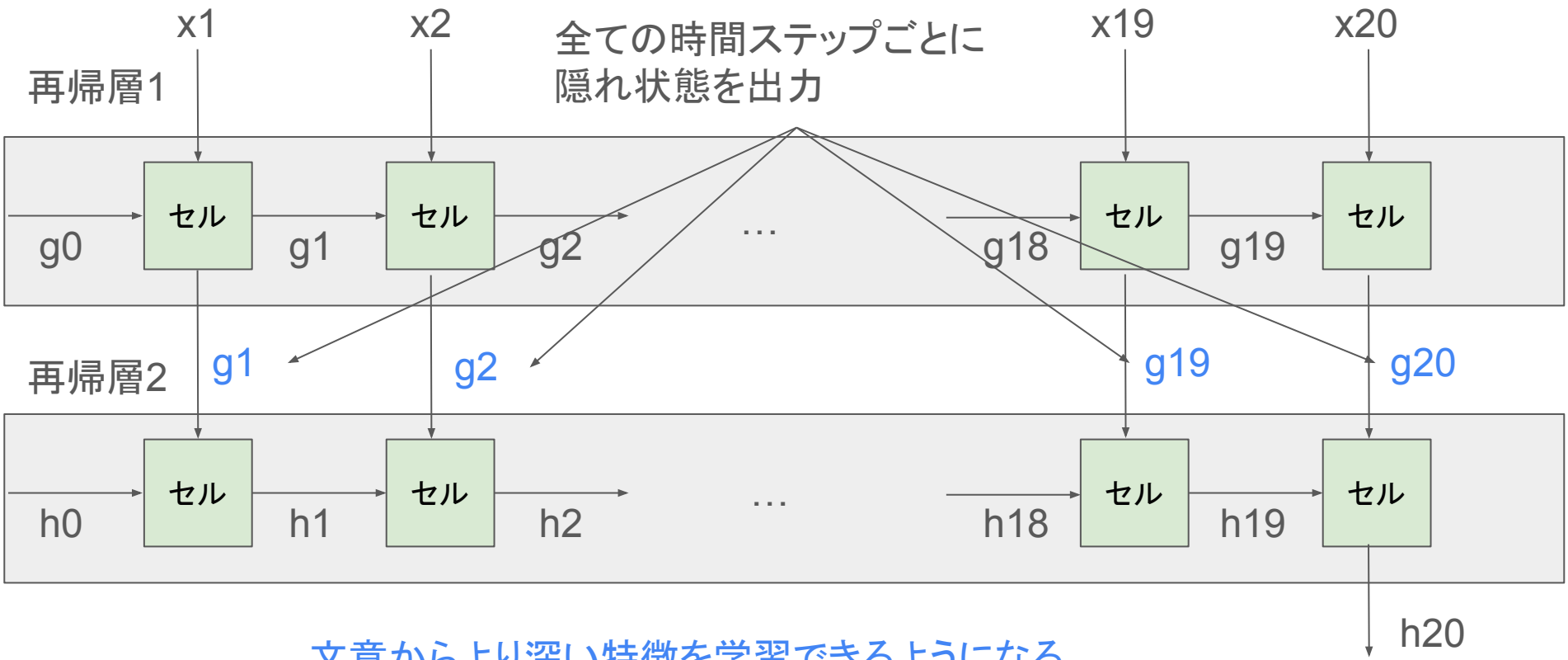
1.0の場合（ちょっと意味が分からなくなってくる）

- the frog and the snake . the frog went to traps on the trap , and managed for sheepfold . when the coast came and swept the have riding he was celebrated , for he had let them to give it in the swimming . just when they were , and they wolves about harmlessly a fee at

5.0の場合（長い単語を多くなってきて更に意味が分からない）

- the frog and the snake . the frog went to withdraw thieving thirsty after nightingale's my flood roundly grapes gain upset cronies mason crowns chaffed curses ingratitude - moment maiden , expression towards flattered dame all flour lie settle with fair habit spoken inquiry number height virtue even health gods ease all sufficiently load away as single foremost

LSTMの多層化



文章からより深い特徴を学習できるようになる

Attention機構、Transformerへと続く