

STUDIO インターン課題メモ

山田 悠

今回の課題について以下の2種類について考えた。

- ① ユーザーが作成したレイアウトを考慮しつつ補正を加える。
- ② ユーザーが作成したレイアウトの強調したい部分を読み取り、それを元により良いレイアウトを提案

今回は計算量の問題から②を実装したが、①についても説明する。説明の簡略化のためにフォントウェイトは700と400の二値を用い、文字列を上から順に t_1, t_2, t_3 とし、文字列の間マージンを p_1, p_2 とした。

また上下のレイアウトによって「よいレイアウト」が変わるため s_0, s_6 は入力値を返した。

まず①で作成しようとしたモデルについて述べる。全体のフローを図1に示す。

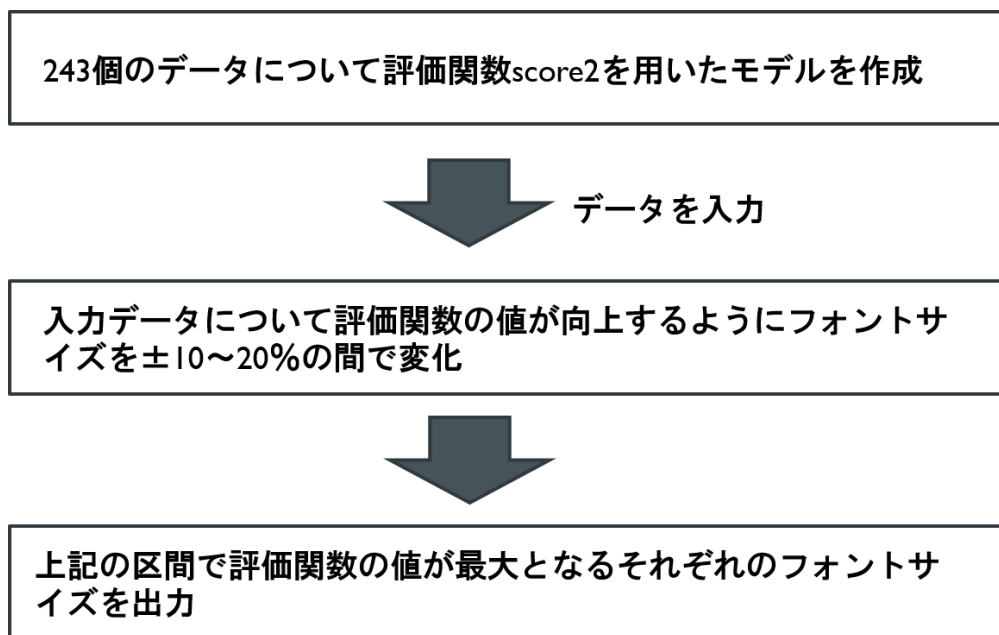


図1 手法①のフロー

具体的な手法について説明する。まず、「よいレイアウト」を数値化するために「readability: 文字の読みやすさ」「highlight: 重要な情報の強調度」「design: デザイン性」の三要素を0~3点で評価し、独自の関数 $score_1$ 、 $score_2$ を作成した。

ここで $score_1$ は三要素の合計値、 $score_2$ はそれぞれの要素について重みをつけた関数で下式に定義した。

$$\text{score2} = \text{readability} + 2 * \text{highlight} + 3 * \text{design}$$

score2の方がscore1よりデザインや強調度の重要な情報が反映されており、大きく分散することから今回はscore2の関数を用いた。次にt1,t2を40から120まで10おきに、t3を30から45まで5おきに变化させレイアウトの評価関数score2のデータセットを作成した。このデータセットは以下の式から324個のデータを持つことがわかる。

$$\left(\frac{120-40}{10}+1\right) \times \left(\frac{120-40}{10}+1\right) \times \left(\frac{45-30}{5}+1\right) = 324$$

この評価関数を用いて、入力データを±10～20%の間で变化させその区間で評価関数が最大となるときの値を出力するモデルを構成することで、ユーザーが作成したレイアウトを考慮しつつ補正を加えることを考えた。

これを実現するには評価関数をt1,t2,t3の重回帰分析で高次の評価関数を得る必要がある。また高次の評価関数のある区間での最大値を求めるには勾配降下法などを用いて計算を行うが、これを行うと3変数であることから計算量が膨大になり即時に出力が行えない問題が発生することが考えられる。この問題点から今回は②のモデルを作成した。

また、1次の式の評価関数をもつ3変数の重回帰分析を行ったが、図2のようになり文字を大きくするほど評価関数の値が向上する(よいレイアウトに近づく)モデルになったためこのモデルは不採用とした。(1次式のため当たり前ではあるが…)このモデルはSTUDIO_jyukaiki.ipynbのファイル名で提出した。

```
[ ] reg = linear_model.LinearRegression()
    reg.fit(X, y)
```

▼ LinearRegression
LinearRegression()

```
▶ print(reg.coef_) #重み
   print(reg.intercept_) #切片
```

```
[0.00222222 0.01736111 0.22864198]
0.1061728395061774
```

図2 重回帰分析の結果

次に今回作成した②のモデルについて述べる。全体のフローを図3に示す。

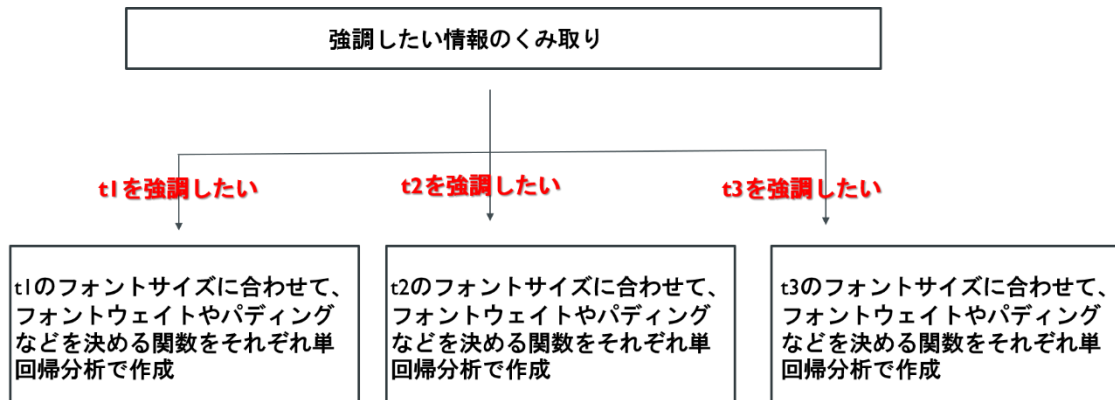


図3 手法②のフロー

このモデルは $t1, t2, t3$ の中から強調したい情報をくみ取り、それを元に良いレイアウトを提案するものである。これを作成するにあたり、まず $t1, t2, t3$ のどの情報を強調したいかに場合分けし、それぞれの場合について $t1, t2$ を 40 から 120 まで 10 おきに、 $t3$ を 30 から 45 まで 5 おきに变化させ最良のレイアウトのフォントサイズの値を保存した。

まず $t1$ が強調したい文字列である場合について考える。 $t1$ と $t2, t3, p1, p2$ の関係調べるためにプロットした。その結果が図4である

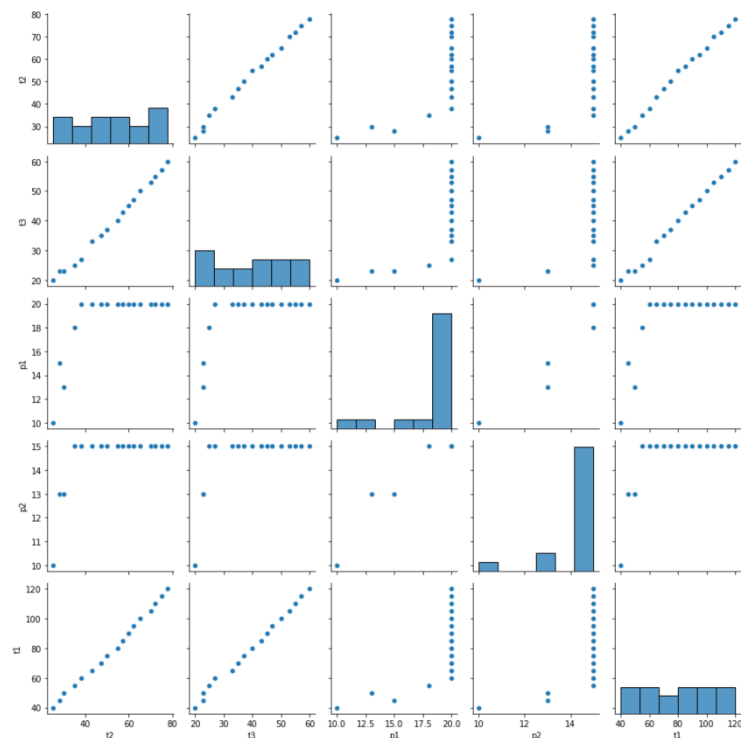


図4 $t1$ とそれぞれの特徴量の関係

図 4 から t_1 と t_2, t_3 の間には一次関数で表される関係があり、 p_1, p_2 にはある値を超えると一定値をとる関係があることがわかる。このことから t_2, t_3 について単回帰分析を行った。

同様にして、 t_3 が強調したい文字列である場合についても分析を行った。ここで t_2 が強調したい文字列の場合については、 t_1 の時と同じ結果を用いたことに留意されたい。これらのモデルを生成したものは `STUDIO_function.ipynb` で提出した。

この手法でそれぞれの関数を生成したものをを用いて作ったものが `STUDIO_coding.ipynb` で提出した。強調したい文字列がない場合(2つのフォントサイズが同一でその他の1つより大きい場合)に関しては、強調したい部分を聞く仕様にした。