





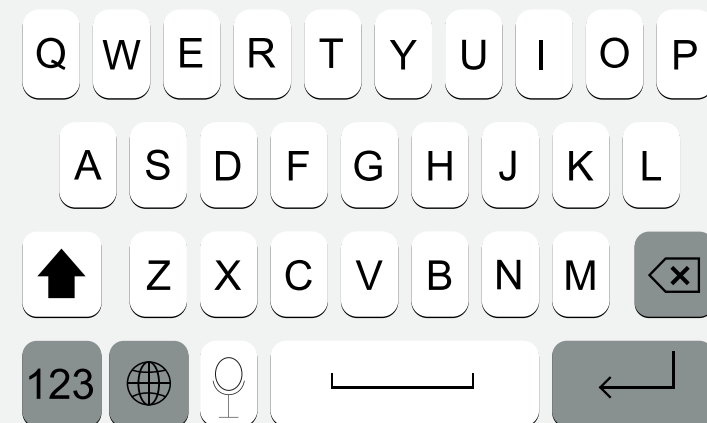
ユーザーに対する キーボード配列の最適化

関西学院大学工学部情報工学課程
37021404 中村 裕大



概要

遺伝的アルゴリズムを活用してユーザーに最適化されたキーボード配列を提供
ユニバーサルデザインの原則に基づいてキーボードを設計し、
指に何らかの障害がある方や、言語による壁を乗り越えることで、
ユーザーの利便性と快適性を向上させることを目的とする



仕組み

5-Step Process

1

使用頻度が
高い単語を
入力する

2

ランダムな
キーボード
配列を
親とする

3

選択法
を用いて
集合群から
要素選択

4

交叉法
を用いて
子の群の
要素生成

5

定数回の
世代数分
繰り返し
結果を出力

遺伝的アルゴリズム

遺伝子の組み合わせと進化の原則を利用して
最適な解を見つけることができるアルゴリズム



以下の手順に沿って最適解を導出する。

1. **初期集団の作成**：ランダムなキーボード配列を生成
2. **適応度の評価**：移動距離が最小となるように評価
3. **選択法**：適応度が高いキーボード配列を選択
4. **交叉法**：選択されたキーボード配列同士を組み合わせで新たなものを作成
5. **突然変異**：ランダム要素を追加し多様性を維持
6. **新世代の形成**：上記の手法を繰り返し新たなキーボード配列を形成



選択法



個体の適応度に基づいて**最適な個体を選び出す**過程

トーナメント選択


親集合の中からランダムに個体を選び出し最も適応度の高い個体を選択

探索の多様性を保ちつつも
優れた個体を選択する手法

エリート選択

適応度の高い親集合の個体を
次世代にそのまま残す方法

解の品質を重視しつつ
探索の収束を抑制する手法





交叉法



選択された親個体から特徴を**組み合わせ**る過程

一様交叉


選ばれた集合要素ごとに
確率的に交換を行う手法

多様性を重視し、
解の探索空間を広げる場合に有効

二点交叉

2つの交点で集合要素を区切り
その範囲内の遺伝子を交換する

特定の特徴や制約を
保持しつつ解を探索



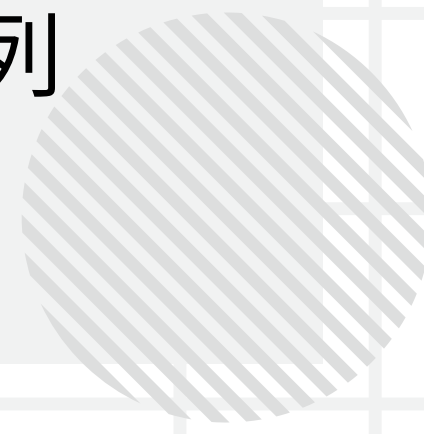
ユニバーサルデザイン

⚙ 指の使用本数 & 使用頻度

本プログラムでは指の使用本数と使用頻度を調整可能にし、ユニバーサルデザインの原則に基づくように設計

⚙ 言語選択

本プログラムではユーザーの使用言語に最適化されたキーボード配列を導出できるように設計





実験



選択法 & 交叉法を変更して
出力されたキーボード配列を観察

実験内容

以下の組み合わせで実験

- エリート選択 & 二点交叉
- トーナメント選択 & 一様交叉

頻出英単語 & 頻出日本語でキーボード配列の比較

- **文章**: "I will show you the SHOYU"(醤油)
- **世代数**: 1000
- **初期世代要素数**: 100
- **指使用頻度**: 人差し指 > 中指 > 薬指 > 小指



実験結果



試行回数	トーナメント形式 & 一様分布	エリート形式 & 二点分布
1	2.5	2.09999
2	2.3	1.5
3	2.09999	1.4
4	2.3	1.5
5	2.5	2.89999
6	2.3	1.5
7	1.7	1.4
8	2.89999	2.2
9	1.7	1.4
10	2.2	2.2
平均値	2.249998	1.809998
中央値	2.3	1.5
最小値	1.7	1.4
最大値	2.89999	2.89999



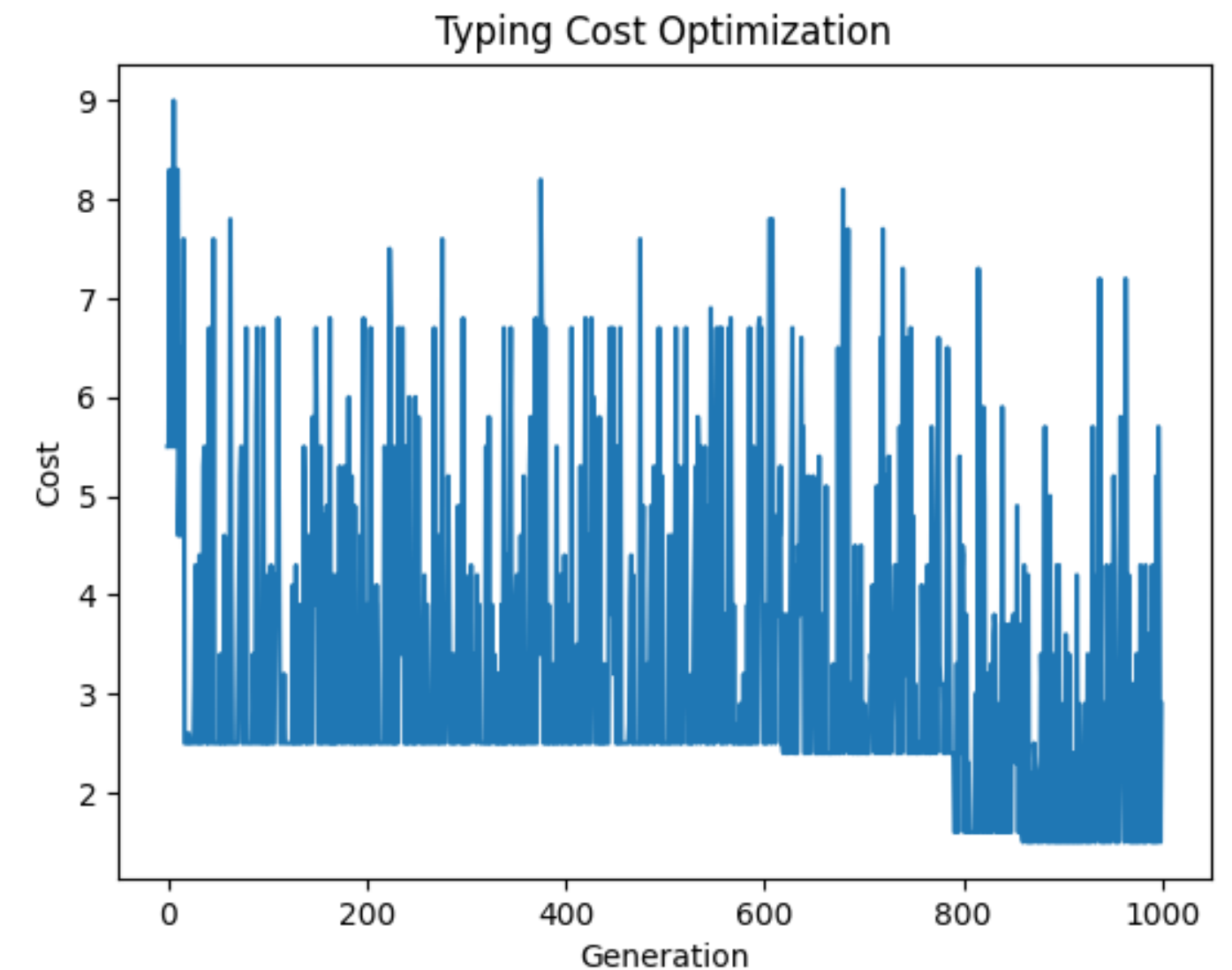
分析結果

エリート選択&二点交叉

導出キーボード配列

R . , P N E T B Q C
Y W S H M Z I L U O
V X K J A D / F G ;

上記のキーボード配列はコスト1.4のものである。ホームポジションを制覇している。グラフより、遺伝的アルゴリズムが正しく作動し、最小コストが更新されている。今後の実験では、こちらを採用。



generation001.py

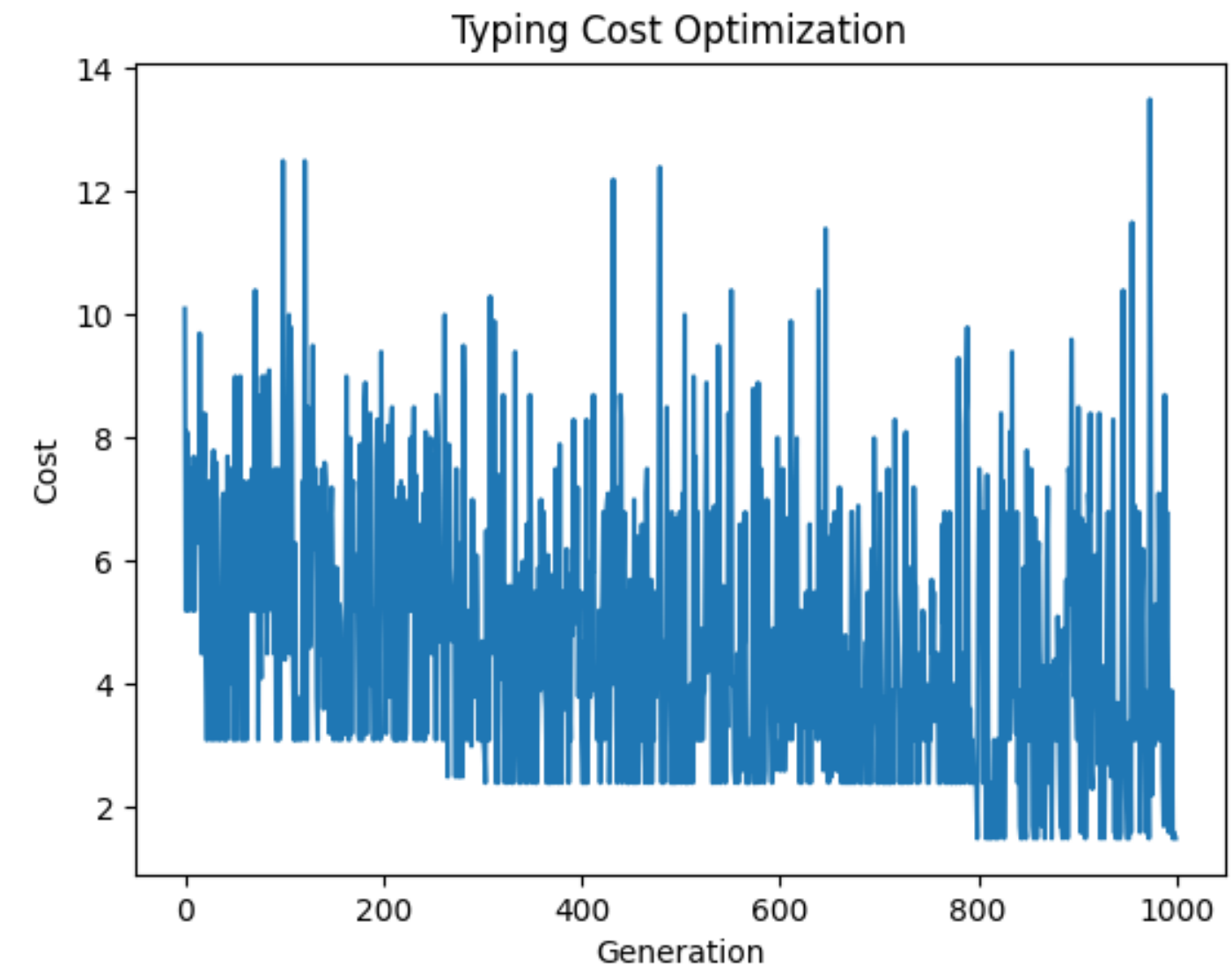
分析結果

トーナメント選択 & 一様交叉

導出キーボード配列

F C X V . G ; N B K
Y U W E A Q L O H I
J R , / Z M T P D S

上記のキーボード配列はコスト1.7のものである。ホームポジション全てが埋まっているので最適解に近いことが分かる。計算時間と最適解導出が遅めである。



generation002.py

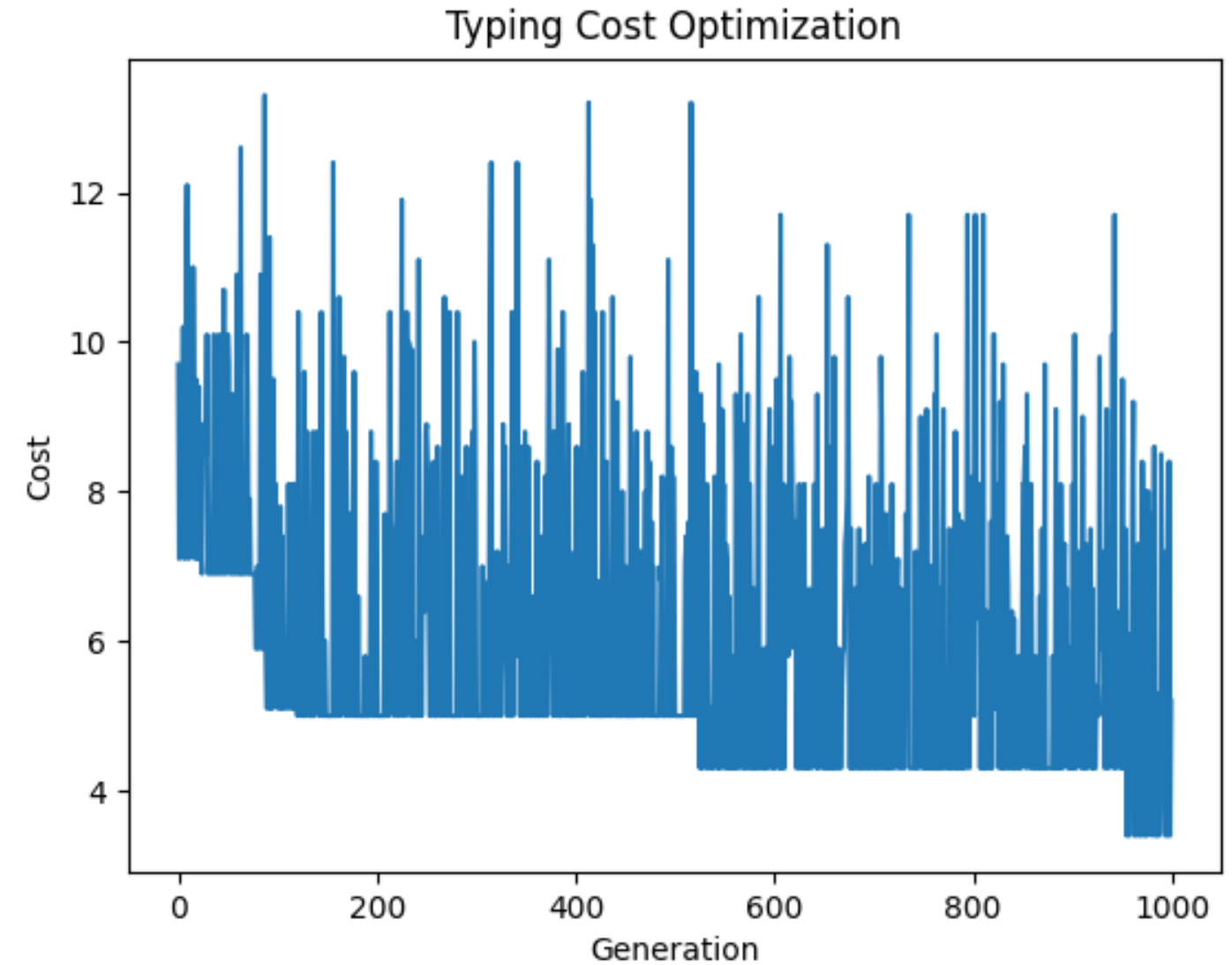
小指使用不可

ユニバーサルデザインの実用例

導出キーボード配列

. X B M N K E T , R
A H Y L S F I W U P
; V G C J Q O / Z D

小指が使用できない場合、ホームポジションの小指を除いた場所の周囲に使用キーが表示される(予想通り)。これは小指が使いやすい人に対してより使いやすいキーボード配列となっていると考えられる。



generation003.py

結果比較



小指使用不可(3.4)

. X B M N K E T , R
A H Y L S F I W U P
; V G C J Q O / Z D

エリート & 二点(1.4)

R . , P N E T B Q C
Y W S H M Z I L U O
V X K J A D / F G ;

トーナメント & 一様(1.7)

F C X V . G ; N B K
Y U W E A Q L O H I
J R , / Z M T P D S

"I will show you the SHOYU"という文章に対して導出方法や
ユニバーサルデザインを考慮することでこんなにも
キーボード配列に違いが出ることが分かった。

エリート & 二点は導出までの時間がかかなり速く実用的であり、
トーナメント & 一様と比較して最小のコストを導き出せる結果となったため、
時間、正確さ共に優れていた**エリート & 二点**を用いるとよいことが分かった。

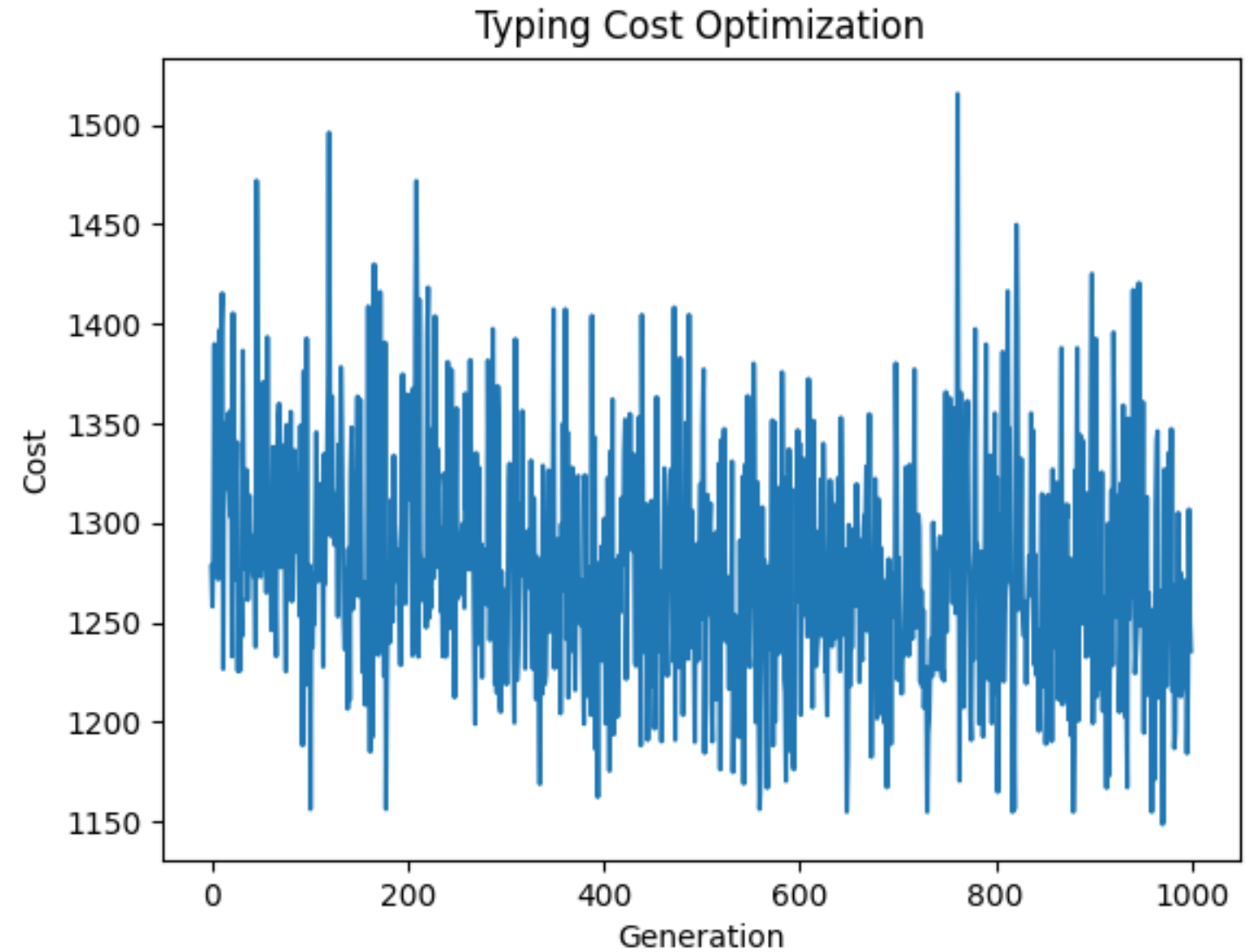
頻出英単語

頻出英単語500を入れるとQWERTY配列に近付くのか

導出キーボード配列

T G H V , ; J P I D
X Y . / W S N A F O
Q E B Z L U K C M R

Levenshtein距離とコサイン類似度を使用して測定した類似度は両者0.0であった。つまりQWERTY配列は英単語を用いる際に適しているとはいえない配列である可能性がある。



generation004.py

頻出日本語

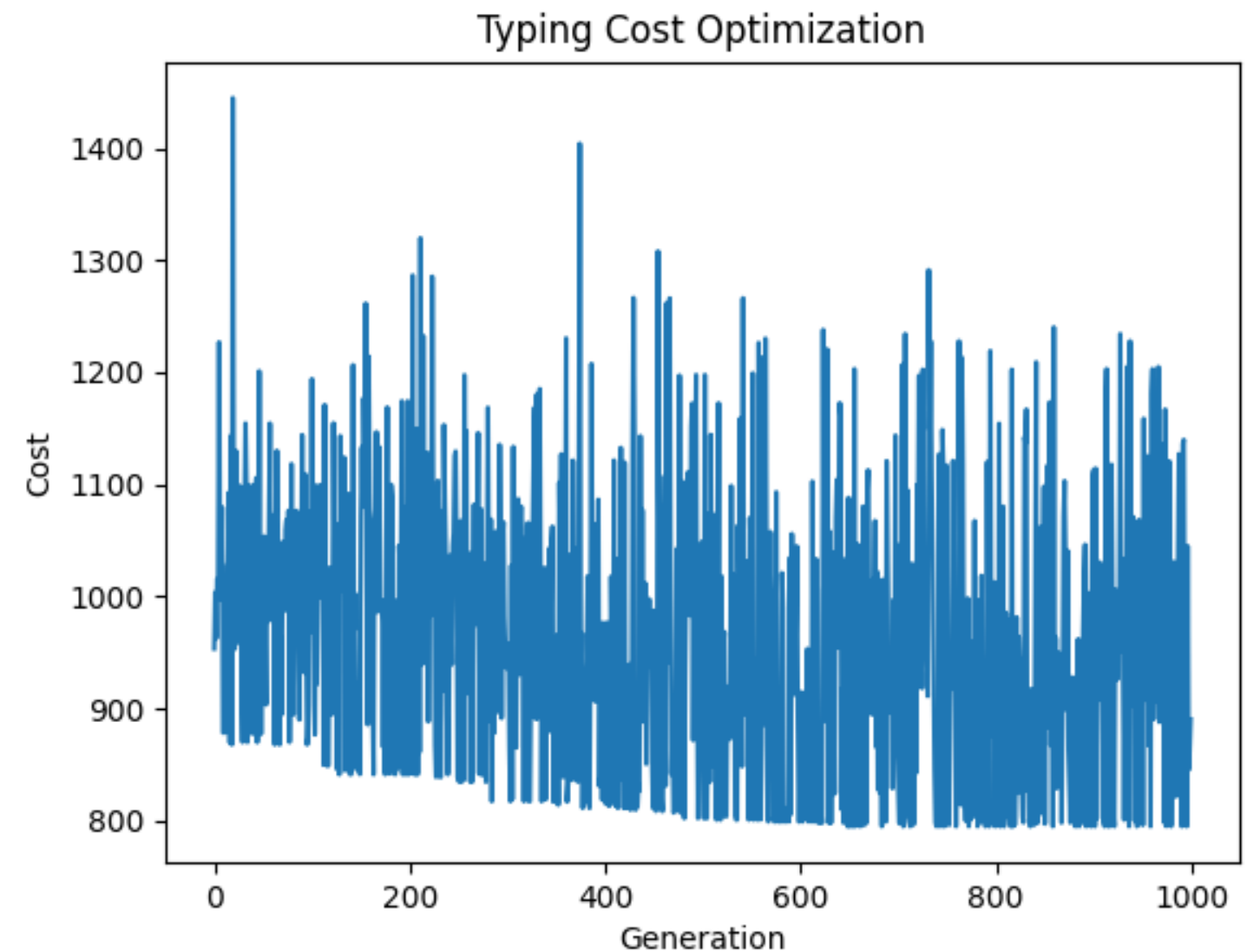
頻出日本語500を入れるとQWERTY配列に近付くのか



導出キーボード配列

S	R	M	T	W	O	V	,	N	X
Z	.	H	D	J	P	F	G	;	I
/	A	C	E	U	Y	B	K	L	Q

こちらも類似度は0.0。日本語で使わないQ, Xなどを除いてもQWERTY配列とは異なるキーボード配列が導出されたという結果になった。



generation005.py



結果比較



QWERTY

Q W E R T Y U I O P
A S D F G H J K L ;
Z X C V B N M , . /

英語

T G H V , ; J P I D
X Y . / W S N A F O
Q E B Z L U K C M R

日本語

S R M T W O V , N X
Z . H D J P F G ; I
/ A C E U Y B K L Q

	英語	日本語
コサイン類似度	0	0
Levenshtein類似度	0	0.1

頻出英単語と頻出日本語を用いてキーボード配列を導出した結果、QWERTY配列とはかけ離れたキーボード配列が誕生した。言語によっても最適なキーボード配列は異なり、ユーザーに対する最適なキーボード配列の導出意義を支える結果となった。

参考文献



🔍 REFERENCES 1

横井浩史(2009)「ソフトコンピューティング演習」
(<http://www.robot.t.u-tokyo.ac.jp/dcm/lec/lec05.pdf>)

🔍 REFERENCES 2

川村インターナショナル(n.d.)「文の類似度を計算する」
(https://www.k-intl.co.jp/blog/B_200729A)



THANK YOU

