

ICS-22M-305

,

卒業論文, 修士論文のタイトル

指導教員 山田 敏規准教授

令和 5 年 12 月 23 日 提出

理工学研究科 数理電子情報系専攻  
情報システム工学コース

22MM305  
安楽岡 花火

埼玉大学 理工学研究科・工学部  
山田研究室  
埼玉県さいたま市桜区下大久保 255

# 概要

本研究では、人工知能 (AI) を用いた新しい癌治療法の開発について検討した。具体的には、AI を用いて癌細胞の特定と分類を行い、その情報をもとに標的療法を行うという方法である。

標的療法は、従来の化学療法や放射線療法と比較して、副作用が少なく、より効果的な治療法として期待されている。しかし、癌細胞の種類や分子構造は多様であるため、標的となる分子を特定することが難しいという課題があった。

本研究では、AI を用いることで、この課題を解決できるのではないかと考えた。AI は、**膨大なデータからパターンを抽出することに長けている**ため、癌細胞の分子構造を特徴付けるパターンを特定することができると考えられる。

本研究では、実際に AI を用いて癌細胞の特定と分類を行うための手法を確立した。この手法を用いて、実際に癌細胞の分子構造を特徴付けるパターンを特定することに成功した。

In this study, we investigated the development of a new cancer treatment using artificial intelligence (AI). Specifically, we proposed a method of using AI to identify and classify cancer cells, and then using that information to perform targeted therapy.

Targeted therapy is a promising treatment that is less likely to cause side effects than traditional chemotherapy or radiation therapy. However, cancer cells are diverse in their types and molecular structures, making it difficult to identify target molecules.

In this study, we hypothesized that AI could be used to address this challenge. AI is **good at extracting patterns from large amounts of data**, so we thought it could be used to identify patterns that characterize the molecular structure of cancer cells.

In this study, we established a method for using AI to identify and classify cancer cells. Using this method, we successfully identified patterns that characterize the molecular structure of cancer cells.

(written by Google Bard)

# 謝辞

本研究の遂行にあたり、多大なるご協力を賜りました関係者の皆様に、深く感謝申し上げます。

特に、本研究の指導をしてくださった [指導教員の名前] 教授に、心よりの謝辞を申し上げます。また、本研究にご協力いただいた [共同研究者の名前] 氏、[共同研究者の名前] 氏にも、厚くお礼申し上げます。

本研究は、[研究費の名称] の助成を受けて行われました。ここに、そのご支援に感謝申し上げます。

We would like to express our sincere gratitude to all those who provided us with great assistance in the conduct of this study.

In particular, we would like to express our deepest gratitude to Professor [Name of the supervisor] for his guidance on this study. We would also like to thank Mr. [Name of the co-researcher] and Ms. [Name of the co-researcher] for their cooperation in this study.

This study was conducted with the support of the [Name of the funding source]. We would like to express our gratitude for their support.

(written by Google Bard)

# 目次

|                       |     |
|-----------------------|-----|
| 概要                    | i   |
| 謝辞                    | iii |
| 図目次                   | v   |
| 表目次                   | vi  |
| 第 1 章 はじめに            | 1   |
| 1.1 背景                | 1   |
| 1.2 目的                | 1   |
| 1.3 本論文の構成            | 1   |
| 第 2 章 序論              | 2   |
| 2.1 typst は優秀だ        | 2   |
| 2.1.1 エレガントに書ける       | 2   |
| 第 3 章 先行研究            | 4   |
| 3.1.1 LATEX はコンパイルが遅い | 4   |
| 第 4 章 定義              | 5   |
| 4.1 定義例               | 5   |
| 参考文献                  | 8   |

# 図目次

|                               |   |
|-------------------------------|---|
| 2.1 サンプル画像 . . . . .          | 3 |
| 3.1 Typst + git [1] . . . . . | 4 |

# 表目次

|              |   |
|--------------|---|
| 2.1 テーブル [1] | 3 |
|--------------|---|

# 第 1 章 はじめに

## 1.1 背景

背景

## 1.2 目的

目的

## 1.3 本論文の構成

構成



## 第 2 章 序論

typst は markdown like なコーディングで pdf, ポスター, スライド等のドキュメントを作成できます. rust 言語で書かれており, コンパイルが latex に比べて早いのが特長です.

### 2.1 typst は優秀だ

こんな感じで @ss8843592 or #cite(<ss8843592>) と引用できます

こんな感じで [2] or [2] と引用できます

#### 2.1.1 エレガントに書ける

`$ mat(1, 2; 3, 4) $ <eq1>`

と書くと

$$a = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \tag{2.1}$$

のように, 式 (2.1) を書くことができます.

また,

```
#figure(  
  image("assets/image.png", width: 20%),  
  caption: [サンプル画像]  
) <img1>
```

とすれば



図 2.1: サンプル画像

Figure 1 を表示できますし,

```
#figure(  
  table(  
    columns: 4,  
    [t], [1], [2], [3],  
    [y], [0.3s], [0.4s], [0.8s],  
  ),  
  caption: [テーブル @madje2022programmable],  
) <tbl1>
```

とすれば

表 2.1: テーブル [1]

|   |      |      |      |
|---|------|------|------|
| t | 1    | 2    | 3    |
| y | 0.3s | 0.4s | 0.8s |

Table 1 も表示できます.

## 第 3 章 先行研究



図 3.1: Typst + git [1]

### 3.1.1 LATEX はコンパイルが遅い

本資料は，LATEX でコンパイルの待ち時間中に作りました．他にも

`#include path.typ`

とすれば，他ファイルを参照できるので，長い分量の本などを作成する際に，章ごとにファイルを分けるなどができるようになります．

便利なので広まれば良いなと思います．

詳しくは 公式ドキュメント をご覧ください

## 第 4 章 定義

Typst では関数定義が簡単であるので定理の書き方などをカスタマイズできます.

### 4.1 定義例

```
#let theorem = thmbox(  
  "theorem", //identifier  
  "定理",  
  base_level: 1  
)  
  
#theorem("すごい定理") [  
  Typst はすごいのである.  
] <theorem>
```

**定理 4.1** (すごい定理): Typst はすごいのである.

定理 1: すごい定理

```
#let lemma = thmbox(  
  "theorem", //identifier  
  "補題",  
  base_level: 1,  
)  
  
#lemma [  
  Tex はさようならである.  
] <lemma>
```

## 補題 4.2: Tex はさようならである.

このように, 定理 1, 補題 2 を定義できます.

カッコ内の引数に人名などを入れることができます. また, identifier を変えれば, カウントはリセットされる. identifier 毎にカウントを柔軟に変えられるようにしてあるので, 様々な論文の形式に対応できるはずです.

```
#let definition = thmbox(  
  "definition", //identifier  
  "定義",  
  base_level: 1,  
  stroke: black + 1pt  
)  
#definition("Prime numbers") [  
  A natural number is called a _prime number_ if it is greater than $1$ and  
  cannot be written as the product of two smaller natural numbers.  
] <definition>
```

定義 4.1: Typst is a new markup-based typesetting system for the sciences.

定義 3 のようにカウントがリセットされています.

```
#let corollary = thmbox(  
  "corollary",  
  "Corollary",  
  base: "theorem",  
)  
  
#corollary [  
  If $n$ divides two consecutive natural numbers, then $n = 1$.  
] <corollary>
```

**Corollary 4.2.1:** If  $n$  divides two consecutive natural numbers, then  
 $n = 1$ .

base に identifier を入れることで Corollary 4 のようにサブカウントを実現できます.

```
#let example = thmplain(  
  "example",  
  "Example"  
) .with(numbering: none)
```

```
#example[  
  数式は \$$\$ で囲む  
] <example>
```

例: 数式は  $\$ \$$  で囲む

thmplain 関数を使って plain 表現も可能です.

```
1: function Binary-Search( $A, n, v$ )  
2:   ▷ Initialize the search range  
3:    $l \leftarrow 1$   
4:    $r \leftarrow n$   
5:  
6:   while  $l \leq r$  do  
7:      $mid \leftarrow \text{floor}(\frac{l+r}{2})$   
8:     if  $A[mid] < v$  then  
9:        $l \leftarrow m + 1$   
10:    else if  $A[mid] > v$  then  
11:       $r \leftarrow m - 1$   
12:    else  
13:      return  $m$   
14:  return null
```

## 参考文献

- [1] L. Mädje, “A Programmable Markup Language for Typesetting”, 2022.
- [2] S. Hussain, S. Bai, and S. Khoja, “Content MathML(CMML) conversion using LATEX Math Grammar (LMG)”, in 2019 7th International Conference on Smart Computing & Communications (ICSCC), 2019, pp. 1–5. doi: 10.1109/ICSCC.2019.8843592.