English LaTeXPPT Template

A simple PPT template for academic

Your name

South China University

April 17, 2024

华南理工大学

South China University of Technology

Catalog

Catalog

Unordered List

- unordered list 1
  - unordered list 1.1
  - unordered list 1.2
  - unordered list 1.3
    - unordered list 1.3.1
    - unordered list 1.3.2
    - unordered list 1.3.3
- unordered list 2
- unordered list 3

Ordered List

1. ordered list 1
    a. ordered list 1.1
    b. ordered list 1.2
    c. ordered list 1.3
        i. ordered list 1.3.1
        ii. ordered list 1.3.2
        iii. ordered list 1.3.3
2. ordered list 2
3. ordered list 3

Description List

word 1 description of world 1

word 2 description of world 2

Catalog

Font

- tiny
- scriptsize
- footnotesize
- normalsize
- large
- Large
- LARGE
- huge
- Huge

- normal
- *italic*
- *slanted*
- **bold**

Catalog

Figure

Figure 1: Logo of SCUT

Subfigure

(a) subfig 1

(b) subfig 2

(c) subfig 3

(d) subfig 4

Figure 2: subfigures

Table

Table 1: Paramter Value

Parameter	Value
α	1
β	1

Table 2: Paramter Value

Module	Parameter	Value
contrastive model	number of RBF centers, $k_{\text{rbf\_c}}$	$\sqrt{n}$
	number of hidden neurons, k_{hidden}	$\frac{\sqrt{n}}{2}$
	dropout rate	0.3
regression model	repetition rate of offline data	10%
	number of centers of one RBFN, $k_{\text{rbf\_r}}$	$\sqrt{\frac{1.1n}{3}}$
topological sorting	threshold thr	$0.3 * nv_{\text{remain}}$
GA	distribution index η_c in SBX	15
	probability of crossover	100%
	distribution index η_m in PM	15
	probability of mutation	$\frac{1}{d}$

You take a screenshot, and throw the picture into the table environment, such as the table on the right.

Catalog

Equation

Interline Formula

$$a_n = a_{n-1} + 1 \tag{1}$$

Inline Formula

This is a simple arithmetic progression formula $a_n = a_{n-1} + 1$.

Catalog

block

title

The content of block. You can use hpsace{2em} to indent two characters at the beginning of the line if the sentence is long.

Abstract

Abstract

The content of the abstract.

Math I

Theorem (title)

body

Lemma (title)

body

proof (title).

body □

Corollary (title)

body

Math II

Example (title)

body

Definition (title)

body

alertblock

title

body

Catalog

Columns

The left side takes up 0.7 width.

The right side takes up 0.3 width.

Catalog

Footnote on Signle Column

It is a simple LaTeX Beamer template for English. Please give me a start on Github [1] if it helps you.

Of course, footnotes can also refer to papers[2].

[1] `https://github.com/h-hg`

[2] K. He, X. Zhang, S. Ren, *et al.*, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

Footnote on Multiple Columns

Footnotes will not be displayed at the bottom of the page by default in the case of multiple columns,such as [a].

Citations for papers[b] are similar.

[a] the default footnote position of multiple columns

[b] K. He, X. Zhang, S. Ren, *et al.*, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

However, it can be solved with a [frame], e.g. [3].

The citation of the paper [4] can be achieved with fullcite.

[3] footntoe with [frame]

[4] K. He, X. Zhang, S. Ren, *et al.*, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778

Catalog

Short Codes

```
#include <iostream>
using namespace std;
int main() {
  cout << "Hello World" << endl;
  return 0;
}
```

You can choose one of the following two methods if the code is long.

Two Columns

One is achieved through the multicols package and changing the font size of the code to put it on the same page of PPT.

```
#include <algorithm>
using namespace std;
void quickSort(int arr[],
               int begin,
               int end) {
  int i, j, t, pivot;
  if (begin > end)
    return;

  pivot = arr[begin];
  i = begin;
  j = end;
  while (i != j) {
    while (arr[j] >= pivot && i < j)
      j--;
    while (arr[i] <= pivot && i < j)
      i++;
    if (i < j)
      swap(arr[i], arr[j]);
  }

  arr[begin] = arr[i];
  arr[i] = pivot;
  quickSort(arr, begin, i - 1);
  quickSort(arr, i + 1, end);
}
```

Multiple Pages I

Another is achieved by allowing the code to span multiple PPT pages.

```
#include <algorithm>
using namespace std;
void quickSort(int arr[],
               int begin,
               int end) {
  int i, j, t, pivot;
  if (begin > end)
    return;

  pivot = arr[begin];
  i = begin;
  j = end;
  while (i != j) {
    while (arr[j] >= pivot && i < j)
      j--;
    while (arr[i] <= pivot && i < j)
```

Multiple Pages II

```
17        i++;
18      if (i < j)
19        swap(arr[i], arr[j]);
20    }
21
22    arr[begin] = arr[i];
23    arr[i] = pivot;
24    quickSort(arr, begin, i - 1);
25    quickSort(arr, i + 1, end);
26  }
```

Catalog

Pseudo-code I

Algorithm 1 KahnAlgorithm

Input: Graph $G(\mathbb{V}, \mathbb{E})$
Output: Sequence L

$L \leftarrow$ an empty sequence
$Q \leftarrow$ the vertices whose indegree is zero
while Q is not empty **do**
    $u \leftarrow$ remove the top node of Q
    add u to L
    for each node v with an edge e from u to v **do**
        remove edge e from graph G
        if indegree of v is 0 **then**
            push v to Q
        end if
    end for
end while
return L

Pseudo-code II

Algorithm 2 Framework

Input: Training data $\mathbb{D}$, Maximum generation $g_{\max}$, Population size n

Output: The best solution

Creating paired dataset $\mathbb{D}_{\text{cl}}$
Training contrastive model M_{con} from $\mathbb{D}_{\text{cl}}$
$i \leftarrow 0$
$P \leftarrow$ Latin hypercube sampling.
while $i < g_{\max}$ **do**
    $C \leftarrow$ apply SBX and PM on P
    $P \leftarrow P \cup C$
    $M_{\text{reg}} \leftarrow \text{BuildRegressionModel}(P, \mathbb{D})$
    $L \leftarrow \text{TopologicalSort}(P, M_{\text{con}}, M_{\text{reg}}, n)$
    $P \leftarrow P[L]$
    $i \leftarrow i + 1$
end while
return $P[0]$

Pseudo-code III

You take a screenshot, and throw the picture into the algorithm environment, such as the algorithm aboved.

Reference I

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

Thanks for your listening!