
基于图卷积神经网络的 复杂网络嵌入方法

姓名 SX17yourID

address@nuaa.edu.cn

2017 年 11 月

摘 要

这是中文摘要。大概写满这一页可以了。摘要又称概要、内容提要。摘要是以提供文献内容梗概为目的，不加评论和补充解释，简明、确切地记述文献重要内容的短文。其基本要素包括研究目的、方法、结果和结论。具体地讲就是研究工作的主要对象和范围，采用的手段和方法，得出的结果和重要的结论，有时也包括具有情报价值的其它重要的信息。

关键词：图卷积神经网络，复杂网络，表示学习

目录

1	引言	1
2	模型	1
2.1	段落	1
2.2	字体	2
2.2.1	文字强调	2
2.2.2	子小节	2
2.3	公式	2
2.4	定理	2
3	实验	2
3.1	图	3
3.2	表	3
4	总结	6

1 引言

这个模板是 UTF-8 编码的，使用 xeCJK 宏包，中英文混排更美观，但编译速度稍慢。注意：该模板只能用 xelatex 编译。本模板支持 Overleaf 在线 \LaTeX 2 ϵ 平台，网址：<https://www.overleaf.com>。详细的 \LaTeX 使用方法这里不详细展开，仅列举一些常用的。

引言也称前言、序言或概述，作为科技论文的开端，提出文中要研究的问题，引导读者阅读和理解全文。

2 模型

正文是指著作的本文，有规范格式和生效标志的正式文本。

2.1 段落

[illegible]

这是下一段。假设这是一个很长的段落，假设这是一个很长的段落，假设这是一个很长的段落，假设这是一个很长的段落，假设这是一个很长的段落，假设这是一个很长的段落，假设这是一个很长的段落，假设这是一个很长的段落，假设这是一个很长的段落。

这是另一段

这又是一段。

这是最后一段，它由好多好多非常多好多非常多好多非常多好多非常多好多好多非常多好多非常多好多非常多好多非常多好多非常多的条目组成，就像下面这个样子：

- [illegible]

很长很长很长很长很长很长很长很长很长很长很长很长的条目。

2.2 字体

默认字体为宋体。**这是黑体**。这是宋体。 这是仿宋。 这是楷体。或者**黑体**，宋体，仿宋，楷体。

2.2.1 文字强调

加粗宋体：**粗体**，加斜字体自动变成楷体：*强调*。

更多中文说明 (网址有点长, 显示不全):

<https://www.overleaf.com/latex/examples/using-the-ctex-package-on-overleaf-zai-overleafping-tai-gndvpvsmjcqx/viewer.pdf>

2.2.2 子小节

这是另一个子小节。

2.3 公式

$$\mathbf{A} = \sum_{i=0}^{N-1} e^{j\frac{2\pi}{N}i} = 0 \quad (1)$$

2.4 定理

正文中可能有很多定理、定义、证明、引理等等。这一特性目前可以手动实现和标号，以后会添加自动方法。

3 实验

[illegible]

存在大量图表。实验中存在大量图表。实验中存在大量图表。实验中存在大量图表。实验中存在大量图表。实验中存在大量图表。实验中存在大量图表。实验中存在大量图表。实验中存在大量图表。实验中存在大量图表。下面演示图表的用法。

3.1 图

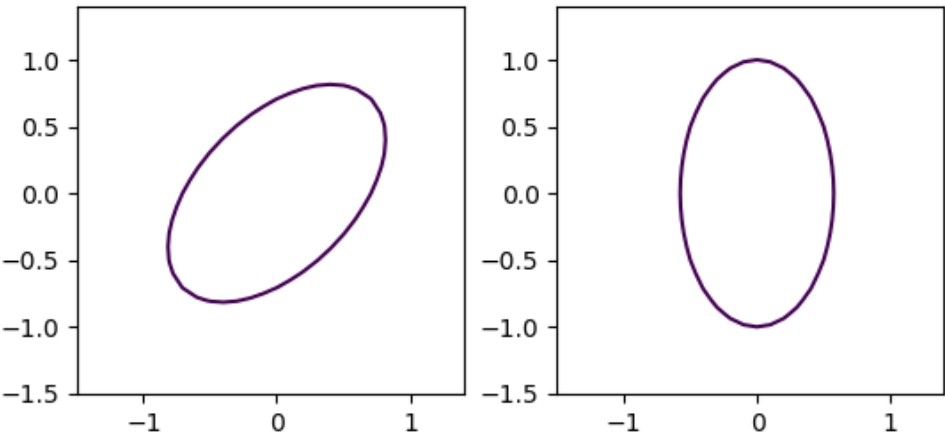


图 1: 这是一个图

引用[1](#);

3.2 表

表 1: 这是一个表

	卡尔曼滤波	神经网络滤波	被动无源滤波
模型类型	线性	线性	非线性
参数调校	大量	几乎没有	合理
稳定性	满足全局稳定性	依赖于模型	满足子系统稳定性

引用表格[1](#)。
代码排版如下。



图 2: 把 SVG 转成 pdf

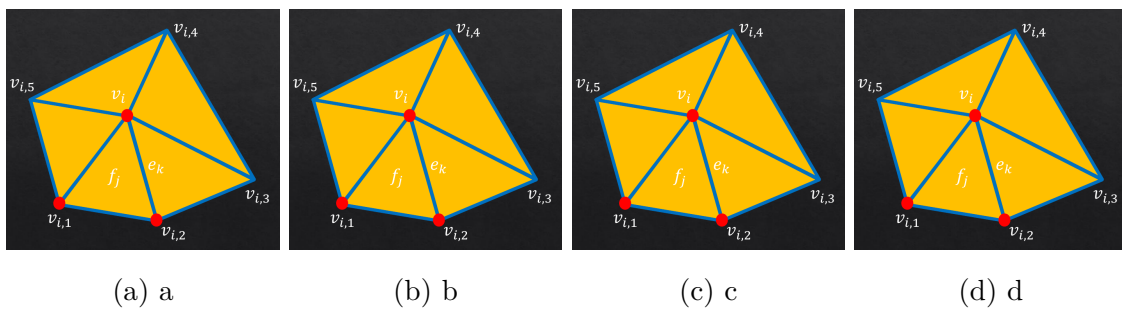


图 3: 横排布局示例

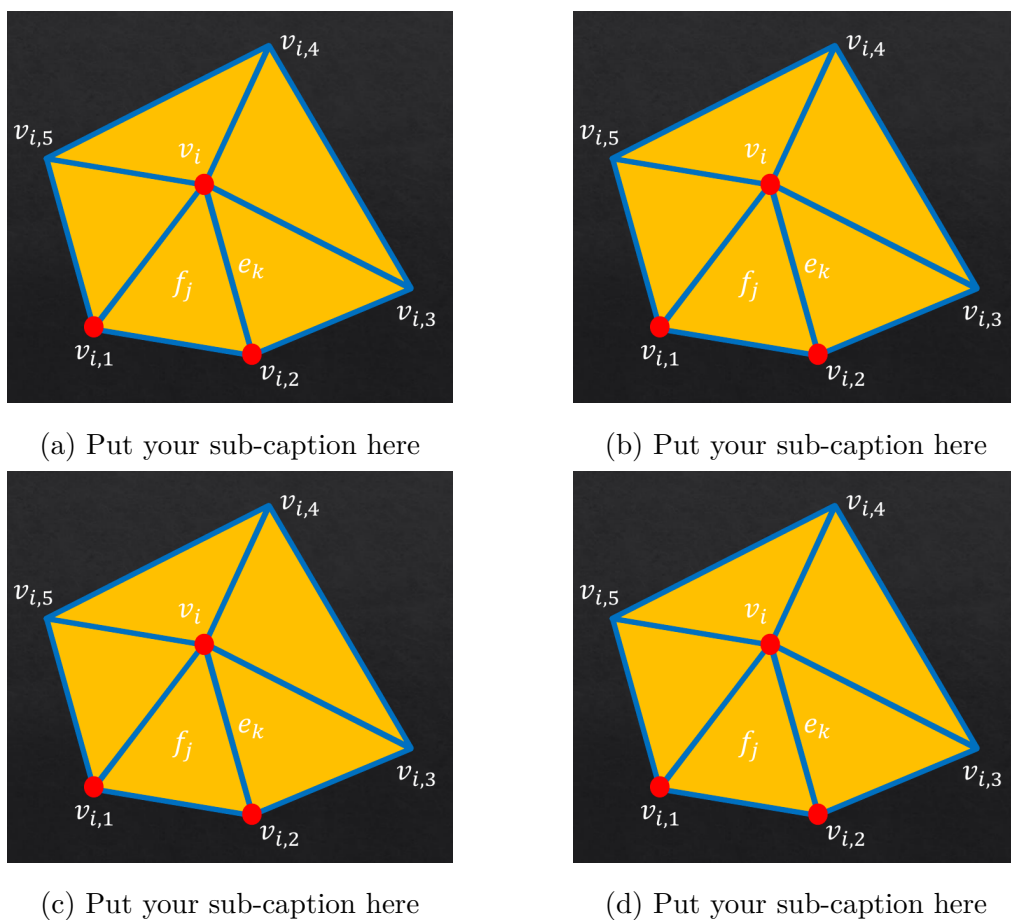


图 4: Put your caption here

```

1  #include <iostream>
2  int main()
3  {
4      std::cout<<"Hello World"!<<std::endl;
5      return 0;
6  }

```

代码 1: Cpp example

```

1  import numpy as np
2
3  def incmatrix(genl1,genl2):
4      m = len(genl1)
5      n = len(genl2)
6      M = None #to become the incidence matrix
7      VT = np.zeros((n*m,1), int) #dummy variable
8
9      #compute the bitwise xor matrix
10     M1 = bitxormatrix(genl1)
11     M2 = np.triu(bitxormatrix(genl2),1)
12
13     for i in range(m-1):
14         for j in range(i+1, m):
15             [r,c] = np.where(M2 == M1[i,j])
16             for k in range(len(r)):
17                 VT[(i)*n + r[k]] = 1;
18                 VT[(i)*n + c[k]] = 1;
19                 VT[(j)*n + r[k]] = 1;
20                 VT[(j)*n + c[k]] = 1;
21
22             if M is None:
23                 M = np.copy(VT)
24             else:
25                 M = np.concatenate((M, VT), 1)
26
27             VT = np.zeros((n*m,1), int)
28
29     return M
30

```

代码 2: Python example

4 总结

这里总结全文。