

2024年度 秋学期

卒 業 論 文

論文タイトル

指導教員: 上原 哲太郎

立命館大学 情報理工学部

卒業研究3 (BA)

コース: セキュリティ・ネットワーク

学生証番号: 26000000000-0

氏名: 立命 太郎

概要

本論文は、サイバーセキュリティ研究室向けの 2023 年度の学士論文用のテンプレートである。あくまでテンプレートであるため、章構成やディレクトリ構成等は自由に変更してよい。また、本論文には、卒業論文執筆を円滑に進めることを目的として、卒業論文執筆におけるアドバイスや注意点を記載している。しかし、取り扱う研究内容によって、論文に執筆する内容が大きく異なることから、アドバイスや注意点は必ずしも守る必要はない。

目次

第 1 章	はじめに	3
1.1	論文執筆にあたって	3
1.1.1	本研究の全体像	3
1.1.2	論文の執筆順序	4
1.1.3	論文の執筆にあたっての注意点	4
1.2	「はじめに」を書く際のアドバイス	5
第 2 章	準備	6
2.1	「準備」を書く際のアドバイス	6
第 3 章	研究背景	7
3.1	「研究背景」を書く際のアドバイス	7
第 4 章	関連研究	8
4.1	「関連研究」を書く際のアドバイス	8
第 5 章	提案手法	9
5.1	「提案手法」を書く際のアドバイス	9
第 6 章	実装	10
6.1	「実装」を書く際のアドバイス	10
第 7 章	評価	11
7.1	「評価」を書く際のアドバイス	11
第 8 章	考察と今後の展望	12
8.1	「考察・今後の展望」を書く際のアドバイス	12
第 9 章	おわりに	13
付 録 A	よく使う TeX 構文	16

第1章 はじめに

品川 [1] や中田 [2] の文献を参考に、各章の書き方をそれぞれの章で説明します。本章では、論文全体の書き方および「はじめに」の書き方について説明します。

1.1 論文執筆にあたって

1.1.1 本研究の全体像

論文をいきなり書くのは大変です。論文を書き始める前に、以下にしたがって研究の大まかなまとめを書いてみてください。

本研究のコンセプト：ポイント・主張したいことは何か？

XXX

本研究の背景：本研究をなぜ研究するのか？

XXX

本研究が取り扱う課題：本研究によって何を解決するのか？

XXX

本研究の提案：どういう方法で課題を解決するのか？

XXX

本研究の貢献点：先行・関連研究と比較して、何が違うのか？

アドバイス：先行・関連研究をまとめて、表 1.1 のようにまとめて比較できると本研究の立ち位置が整理がしやすいです。

手法	観点 A	観点 B
既存手法 1 (引用)	△	×
既存手法 2 (引用)	○	×
提案手法	◎	◎

表 1.1: 既存手法と本研究の提案手法の比較

1.1.2 論文の執筆順序

論文を「はじめに」から順に書くのは大変です。そのため、研究の全体像を把握した上で、以下の順序で論文を書くことをおすすめします。無論、この順序に従う必要はありません。

「謝辞」はウォーミングアップとして、最初に書いてみましょう。次に、「研究背景」や「関連研究」など、すでに自分の中で整理できている部分の執筆と並行して、「実装」や「評価」など、まだ論文に書くに足らない部分の進捗を出しましょう。

1. 謝辞
2. 研究背景・関連研究
3. 実装・評価
4. 考察と今後の展望
5. はじめに・おわりに
6. 概要

1.1.3 論文の執筆にあたっての注意点

重要なこと

- **原稿のタイポや誤字脱字、凡ミスを普段からなるべく減らす努力をしましょう。** 多いほど先生から本質的なコメントをもらいづらくなります。先生が使える時間は有限なので、些末なミスばかりだと深く読み込むことが難しく、少しずつ修正が進んで読める原稿になってきた段階になるころには締切が近づいており、突然ボコボコに修正する羽目になってお互い辛い目に遭います。こうならないためには、まず自分自身で、前述の書く順を守ること、Word に文をコピペしてみてエラーをチェックすること、Cylint [3] を用いてチェックすることを実施してみてください。その後、先輩や同期など学生に見てもらい、最後に先生に見てもらおうという流れが良いと思います。
- **複数の人に見てもらいましょう。** 一人にいっぱい見てもらうのは大変なので、「はじめに」の最初だけ、「概要」だけというように絞って意見をもらうのも手です。(ほぼ査読のレベルで頑張って読んでコメントをくれた人たちは謝辞に入れた方が良いので、名前をメモしておきましょう。)
- **バックアップをとりましょう。** なぜか卒論時期に限ってマシンの故障が頻発します。提出間近になって PC が壊れてしまい、提出できなくなるという悲惨な事故は起こり得ます。そうならないように、ローカル環境だけでなく OneDrive や Google Drive, GitHub などのクラウドサービスにバックアップ体制を構築してお

きましょう。バージョン管理が可能であるため、誤った更新やデータ損失が発生した場合に簡単に元の状態に戻せることや、GitHub の Issue や Pull Request, Permalink 機能を用いたレビューができることから、サイバーセキュリティ研究室では、GitHub の利用を推奨します [4].

- **時間を決めて毎日書きましょう。** 論文を一度にがっと思書くのは大変ですし、ペースもつかみにくいです。30 分で時間を区切ったり、「今日は関連研究の一つ目を埋めよう」「図を描こう」など、短期的な目標を一つずつ設定すると執筆の困難さを低減することができます。(自分の文章自体をついでにチェックしたりもついついするので、タイポや誤字脱字の修正機会を増やせます)

気をつけると良いこと

- 使用用語はなるべく統一しましょう。紛らわしいものは明確な線引きをする意識で書きましょう。同じ単語なのに、違う意味を持つことや、同じ意味を持つのに違う単語を使っていると、読み手が混乱します。
- 主語を省略しないようにしましょう。省略しすぎると、読み手がなにを指しているのか分からなくなります。そのほかにも論文では適さない表現がいくつかあります。見延 [5] がまとめているので、参考にしてください。
- 句読点「,」(全角カンマピリオド)を使う時は注意してください。半角で入力されていると、後から一括変換しようにも思わぬ場所まで変換されてしまう羽目になります。

原稿を見てもらったら

論文の原稿添削で同じことを繰り返し指摘する側も大変ですが、指摘される側もまあ辟易とするかと思います。今後同じ目に合わないよう、復習をしましょう。おすすめは原稿ごとに指摘された内容をメモしておいて、執筆が終わったらメモを見返して、ここの編集要求の意図は何だったのかを考えたり、どうすれば指摘を受けなくて済むのか考えてみるのがオススメです。論文の書き方の本を片手に持ちながら、疑問を解消するのもぜひやってみてほしいと思います。執筆の終わった直後が、一番学習効率が良いです。

1.2 「はじめに」を書く際のアドバイス

- 書き出しを「近年」ではじめるのは、読み手によって近年が示す時間軸が異なるため禁止。何を実現する上で何が重要なのか、分野において何が重要なのかを本研究のコンセプトに紐づけて一言で書いてみましょう。
- 各段落の最初の一文だけをつなげておおよその流れが理解できるように書きましょう。つまり、まず各段落の最初の一文だけを書いてストーリーが通っているかを先生と一緒に確認し、OK が出たら各段落に詳細な内容を肉付けしていく、という手順で書いてみてください(これは「はじめに」に限らず、原稿全体で同様の手順で進めるのが理想的です)。

第2章 準備

本章では、本研究で扱う用語の中で、読者にとって理解しにくいと筆者が考える用語について、説明しましょう。ただし、筆者が他の章において論文中の用語を説明しており、かつ読者が理解しうると判断した場合、本章は必ずしも必要ではありません。

2.1 「準備」を書く際のアドバイス

- 本論文において重要な用語については、「研究背景」や「関連研究」で説明することが望ましいです。
- 「準備」はあくまで、本論文で用いる用語や概念等の説明を補助することが目的であるため、本論文と強く関連のある用語や概念等については他の章で説明することが望ましいです。

第3章 研究背景

本章では、なぜ本研究に取り組もうと思ったのか、本研究に取り組む意義について、社会的背景等に基づいて説明しましょう。

3.1 「研究背景」を書く際のアドバイス

- なぜ本研究に取り組もうと思ったのか、本研究に取り組む意義について、説明するにあたって必要な背景を節ごとに書いてみましょう。
- 時代の流れが本研究に強く関連している場合は、時系列順に書くことを意識してみてください。

第4章 関連研究

本章では、本研究を先行研究と比較してどのように位置づけることができるか（どのような流れに乗っていて、何が新規な点であるか）を説明しましょう。

4.1 「関連研究」を書く際のアドバイス

- 「関連研究」は、本研究の位置づけを説明するためにあります。本研究との関連を述べずに紹介されている関連研究は存在意義がありませんので、関連を述べることを意識して書いてみてください。
- 何を関連研究として載せるかの選択は重要です。大抵は、分野、問題、解決手段に分けることが多いですが、基本的には、本研究のウリ・面白い点を際立たせるにはどのような観点で述べると良いかを考えると良いと思います。
- 必ずしも綺麗に時系列順に並んでいることが望ましいとは限りません。話が行ったり来たりしないように、トピックごとにまとめて書くことを意識してみてください。

第5章 提案手法

本章では，研究背景・関連研究を踏まえて，どのように課題を解決するのかについて書きましょう．

5.1 「提案手法」を書く際のアドバイス

- なにをすれば，どのようなことがわかる or できるようになるのかを書きましょう．
- また，わかったこと・できたことがどう課題解決につながるのかを書きましょう．
- 使用したツールなどは，実装で書くためこの章では，大まかな理論や手法を書きましょう．

第6章 実装

本章では，なにを使って提案手法を実現したのかについて書きましょう．

6.1 「実装」を書く際のアドバイス

- 本章では，提案手法を実現するために使用したツールや，作成したものについて説明しましょう．
- 「提案手法」の構成と対応づけて，「実装」を書くと読者がとても読みやすくなります．
- 卒業論文は，結果よりも過程が評価対象となるため，使用した言語やツールのバージョン，パラメータ，ロジック等をできるだけ詳細に記述しましょう．

第7章 評価

本章では，本研究の目的がどの程度達成されたのかについて説明しましょう．

7.1 「評価」を書く際のアドバイス

- 「はじめに」で明らかにする・示すと宣言した内容に対して，一つずつ対応させながら評価項目を書くことを意識してみましょう．
- 評価を行うにあたって，あらかじめ自分で設定した仮定については，事前に明らかにしておきましょう．
- 「結果」という章を作って，「評価」の結果をまとめて書いても構いません．

第8章 考察と今後の展望

本章では、評価結果を総合的に見て、知見として主張できる内容と、評価結果が示唆している内容をまとめましょう。また、考察をもとに、本研究における課題・展望を重要な課題に絞って述べましょう。

8.1 「考察・今後の展望」を書く際のアドバイス

- 「～だと思う」のように一人称視点で書くのではなく、第三者的視点で「～だといえる」「～だと示唆される」「～だと思われる」と書くようにしましょう。これはつまり、個人の感想ではなく、「結果からどう考えてもそう結論付けざるを得ない」という第三者的な目線で書くように意識して書いてみてくださいということです。
- 結果から強く主張できる場合は「～だといえる」、エビデンスが不十分な場合は「～だと示唆される」「～だと思われる」を使いましょう。
- 客観的事実と著者の主張が混ざらないように気をつけましょう。
- 本研究の限界についても述べましょう（課題は挙げるときりがなく、述べ過ぎると具合が悪いので、重要な課題に絞って「これがまだできてなくて今後解決する必要がある」と説明する）。

第9章 おわりに

本研究が解決しようとした問題とは何か，そしてその問題を（提案手法で）どのようにして解決しようとしたのか，実験の結果から何が主張でき，本研究の貢献として何が新たに知見として得られたのか，本研究の限界・残された課題は何か，今後どのように研究を展開していく予定かを述べます．ここだけは，著者の本研究におけるアツい思いについて一言述べても OK （例：「本研究が〇〇における重要な一歩になると期待している」とか）．

謝辞

本章では，本研究を進めるにあたってお世話になった方々への感謝の気持ちを述べましょう．以下に，立命館大学の穂山先生の言葉を引用します．

謝辞は決まり文句で済ましてしまいがちであるが、それはよくない。これまで支えてくれた人たちのことを一旦落ち着いてよく考えてみて欲しい。ちなみに穂山の博論の謝辞は高校以来お世話になった友人をすべて挙げている。これを真似せよという意味ではなく、各自が自分なりの感謝の気持ちをしっかり認識することが重要である。

参考文献

- [1] 品川政太郎. 「頑張らない学生指導」を頑張ろう. 入手先 <https://snowman-88888.hatenablog.com/entry/2023/12/30/215616> . (参照 2024-1-03) .
- [2] 中田亨. 卒業論文の書き方. 入手先 https://researchmap.jp/blogs/blog_entries/view/77082/0eb9a69b5e4a770751c9f9b0f7f315d0 . (参照 2024-1-03) .
- [3] 石川琉聖. Latex linter. 入手先 <https://cylint.ryuse.dev> . (参照 2024-1-03) .
- [4] cysec lab. 卒論執筆 指南書. 入手先 <https://github.com/cysec-lab/CyTeX/blob/feature/bachelor-2024-procedure-manual/procedure-manual.md> . (参照 2024-1-03) .
- [5] 見延庄士郎. 卒・修論に見られる良くない表現. 入手先 https://www.sci.hokudai.ac.jp/~minobe/class/bad_expressions.htm . (参照 2023-12-05) .
- [6] 日本語 TeX 開発コミュニティ. Tex wiki. 入手先 <https://texwiki.texjp.org/> . (参照 2023-12-05) .

付 録 A よく使う TeX 構文

よく使う TeX の記述例を示す. 詳しくは, TeX Wiki [6] を参照されたい.

- `itemize1`
 - `itemize2`
 - `itemize3`
1. `enumerate1`
 2. `enumerate2`
 3. `enumerate3`

図 A.1, 図 A.2a, 図 A.2b にテツ太郎ロゴを示す.

表 A.1 に表の例を示す.

ソースコードの例を ListingA.1 に示す.

また, `listings` パッケージが対応している言語を図 A.3 に示す [?].



図 A.1: テツ太郎ロゴ



(a) テツ太郎ロゴ 2



(b) テツ太郎ロゴ 3

図 A.2: テツ太郎ロゴ

表 A.1: 表の例

	column1	column2	column3
row1	item 1,1	item 2,1	—
row2	—	item 2,2	item 3,2
row3	item 1,3	item 2,3	item 3,3
row4	item 1,4	item 2,4	item 3,4

Listing A.1: ソースコードの例

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 x = np.linspace(0, 2*np.pi)
5 y = np.sin(x)
```

ABAP (R/2 4.3, R/2 5.0, R/3 3.1, R/3 4.6C, R/3 6.10)	
ACM	ACMscript
ACSL	Ada (<u>2005</u> , 83, 95)
Algol (60, <u>68</u>)	Ant
Assembler (Motorola68k, x86masm)	Awk (<u>gnu</u> , POSIX)
bash	Basic (<u>Visual</u>)
C (<u>ANSI</u> , Handel, Objective, Sharp)	
C++ (11, ANSI, GNU, <u>ISO</u> , Visual)	Caml (<u>light</u> , Objective)
CIL	Clean
Cobol (1974, <u>1985</u> , ibm)	Comal 80
command.com (<u>WinXP</u>)	Comsol
csh	Delphi
Eiffel	Elan
elisp	erlang
Euphoria	Fortran (03, 08, 18, 77, 90, <u>95</u>)
GAP	GCL
Gnuplot	Go
hansl	Haskell
HTML	IDL (empty, CORBA)
Inform	Java (empty, <u>AspectJ</u>)
JVMIS	ksh
Lingo	Lisp (empty, Auto)
LLVM	Logo
Lua (5.0, 5.1, 5.2, <u>5.3</u>)	make (empty, <u>gnu</u>)
Mathematica (1.0, <u>11.0</u> , 3.0, 5.2)	Matlab (empty, 5.1)
Mercury	MetaPost
Miranda	Mizar
ML	Modula-2
MuPAD	NASTRAN
Oberon-2	OCL (<u>decorative</u> , <u>OMG</u>)
Octave	OORexx
Oz	Pascal (Borland6, <u>Standard</u> , XSC)
Perl	PHP
PL/I	Plasm
PostScript	POV
Prolog	Promela
PSTricks	Python (<u>2</u> , 3)
R	Reduce
Rexx (empty, VM/XA)	RSL
Ruby	S (empty, PLUS)
SAS	Scala (empty, 3.0)
Scilab	sh
SHELXL	Simula (<u>67</u> , CII, DEC, IBM)
SPARQL	SQL
Swift	tcl (empty, tk)
TeX (AlLaTeX, common, LaTeX, <u>plain</u> , primitive)	
VBScript	Verilog
VHDL (empty, AMS)	VRML (<u>97</u>)
XML	XSLT

図 A.3: listings パッケージが対応している言語