

UNDERGRADUATE PROJECT REPORT

|  |  |
| --- | --- |
| **Project Title:** | **Book Recommendation Using Deep & Cross Network** |
| **Surname:** | **Cheng** |
| **First Name:** | **Arnold** |
| **Student Number:** | **201918020422** |
| **Supervisor Name:** | **Joojo Walker** |
| **Module Code:** | **CHC 6096** |
| **Module Name:** | **Project** |
| **Date Submitted:** | **May 5, 2023** |

# **Declaration**

Here, students would sign a statement indicating that they adhered to appropriate academic conduct in carrying out their final project.

# **Acknowledgment**

Here, students are given the opportunity to thank those who have provided you with assistance and support.

# **Table of Contents**

[**Declaration** i](#_Toc129605831)

[**Acknowledgment** ii](#_Toc129605832)

[**Table of Contents** iii](#_Toc129605833)

[**Abstract** iv](#_Toc129605834)

[**Abbreviations** v](#_Toc129605835)

[**Glossary** vi](#_Toc129605836)

[**Chapter 1 Introduction** 1](#_Toc129605837)

[**1.1** **Background** 1](#_Toc129605838)

[**1.2** **Aim** 1](#_Toc129605839)

[**1.3** **Objectives** 1](#_Toc129605840)

[**1.4** **Project Overview** 1](#_Toc129605841)

[**1.4.1** **Scope** 1](#_Toc129605842)

[**1.4.2** **Audience** 1](#_Toc129605843)

[**Chapter 2 Background Review** 2](#_Toc129605844)

[**Chapter 3 Methodology** 3](#_Toc129605845)

[**3.1** **Approach** 3](#_Toc129605846)

[**3.2** **Technology** 3](#_Toc129605847)

[**3.3** **Project Version Management** 3](#_Toc129605848)

[**Chapter 4 Results** 4](#_Toc129605849)

[**Chapter 5 Professional Issues** 5](#_Toc129605850)

[**5.1** **Project Management** 5](#_Toc129605851)

[**5.1.1** **Activities** 5](#_Toc129605852)

[**5.1.2** **Schedule** 5](#_Toc129605853)

[**5.1.3** **Project Data Management** 5](#_Toc129605854)

[**5.1.4** **Project Deliverables** 5](#_Toc129605855)

[**5.2** **Risk Analysis** 5](#_Toc129605856)

[**5.3** **Professional Issues** 5](#_Toc129605857)

[**Chapter 6 Conclusion** 6](#_Toc129605858)

[**References** 7](#_Toc129605859)

[**6.1** **Formatting Requirements** 7](#_Toc129605860)

[**6.2** **Written Presentation** 7](#_Toc129605861)

[**Appendices** 8](#_Toc129605862)

# **Abstract**

Up to 250 words, concise outline of background, aims, results, and achievements.

***Keywords: This is part of the “Abstract” section. Students are to provide at least 3 keywords that best describe their project topic. Do not list more than 8 keywords.***

# **Abbreviations**

This section should have the definition of any abbreviations used in the report

# **Glossary**

This section should have the definition of all the keywords you stated in the “Abstract” section. You can also define other relevant keywords. Particularly, if your final project report includes rare, unfamiliar, specialized, or made-up words or terms, the glossary serves as a dictionary for the reader to reference throughout their reading of the project report. (Note: this section should only contain definitions for specific terms in the project report. It does NOT function as an ordinary dictionary. Hence, common words related to the Computer Science and Software Engineering disciplines should NOT be included in this list.)

# **Introduction**

*This is an update of the Introduction from your previous reports. Students are to also incorporate the feedback from their supervisor. Also, include subsequent ideas and research that you have discovered.*

## **Background**

This section should describe the overview of the topic and motivations. Provide appropriate references wherever necessary.

## **Aim**

The overall goal of your project should be stated here. It is recommended that each project should have a single aim.

## **Objectives**

Students are to state the several tasks/steps that would help them to accomplish the overall aim/goal of their project.

## **Project Overview**

(NB: Most students are working either on a software development-based project or a machine learning/deep learning-based project. Hence, in section 1.4, students must adopt the appropriate theme/content depending on their project topic.)

### **Scope**

The scope of a software development project should answer the questions: what will the software do? How will the software work? The scope for a machine learning/deep learning-based project should focus on answering the questions: what is the purpose of the study? How significant is the study?

### **Audience**

The audience for a software development project should focus on who is the software for? The audience for a machine learning/deep learning-based project should focus on who will benefit from the findings.

# **Background Review**

This chapter is an update of the Background Review from your previous reports, using the feedback you received from your supervisor. Compare existing approaches and include a themed literature review, with a critical appraisal of the sources. Provide appropriate and sufficient references. Also include *the feedback you received from your supervisor. You can add any additional key sources that you have discovered.*

Students doing software development-based projects can write their background review by providing a **summary of existing approaches (e.g., competitive analysis, if appropriate),** and others doing research-oriented projects (machine learning & deep learning projects) can write their background review by stating **a summary of related literature (e.g., annotated bibliography, or initial literature review, with a brief summary of sources).**

**Annotated Bibliography aids as in doing a good literature review. It is not the literature review. However, your final background review must be paragraphs with appropriate citations. Whenever appropriate, a table can be adopted.**

# **Methodology**

## **Approach**

A brief description of the Deep & Cross Network model

The Deep & Cross Network model we will refer to as the DCN model below:

In this project, the DCN (Deep & Cross Network) model is employed as the core recommendation model for the proposed solution. The complete DCN v1 model is shown in the figure:

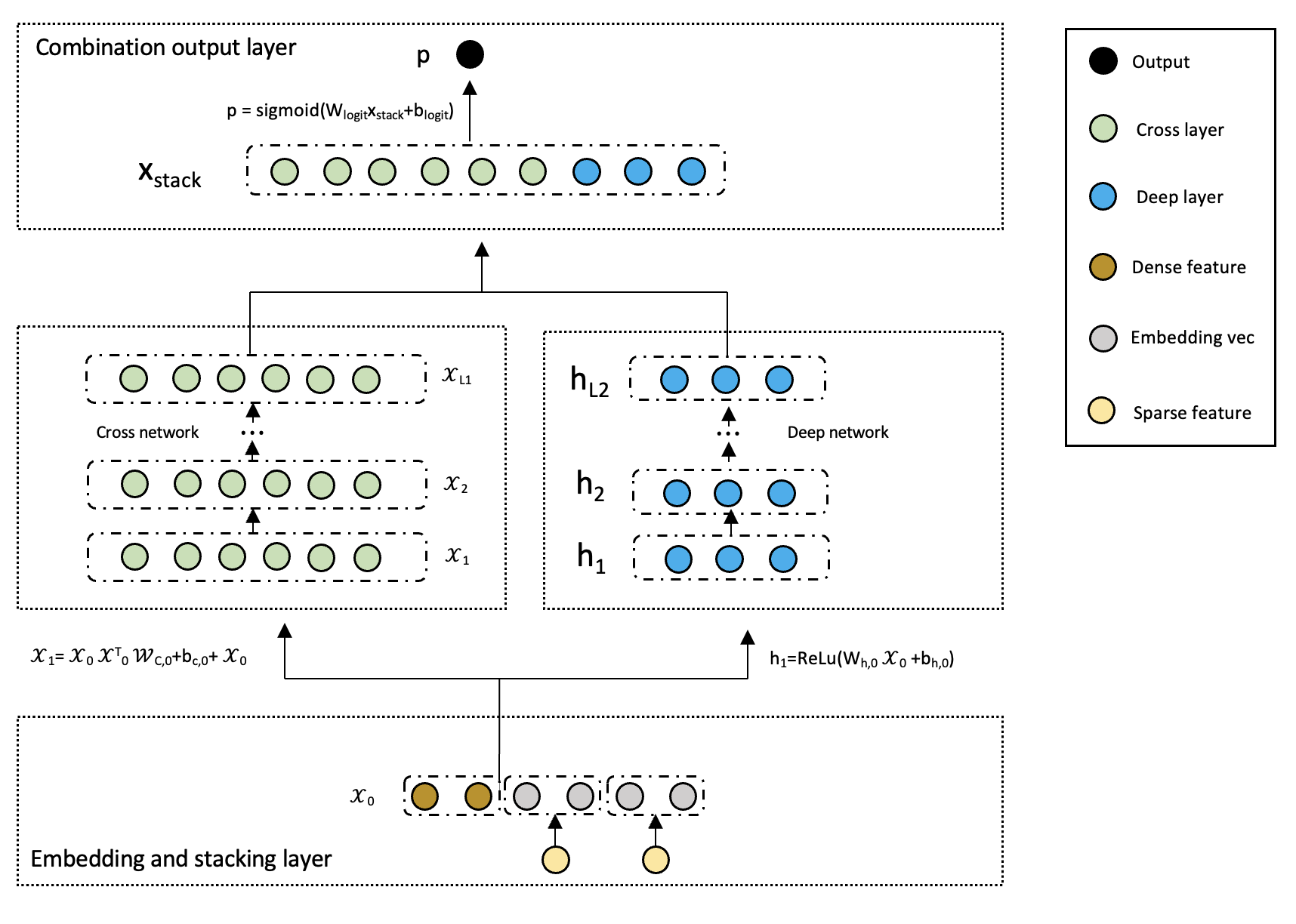


Figure 1: DCN v1 Model

DCN can be divided into four main parts. The first part, located at the bottom, is the "Embedding and Stacking Layer". This layer involves the process of converting discrete features into dense vectors using an embedding technique, and then stacking these embedding vectors with the continuous features to form a unified input vector for the subsequent layers.

When dealing with input data that exhibit both discrete and continuous characteristics, such as in network-scale recommendation systems like click-through rate (CTR) prediction, the input is typically comprised of categorical features, such as "country=USA". These categorical features are frequently represented using one-hot encoding, resulting in a binary vector such as "[0,1,0]". However, this approach often results in a high-dimensional feature space when dealing with large vocabularies.

To reduce the dimensionality, the embedding process is used to convert these discrete features into a dense vector of real values (often called an embedding vector):

= (2)

Then, superimpose the embedding vector with the continuous eigenvector to form a vector:

= (3)

The stitched vector 0 will serve as input to our Cross Network and Deep Network

The second and third parts are the "Cross Network" and the "Deep Network", respectively, which are in the middle of the architecture. The Cross Network utilizes explicit feature interactions to capture the pairwise correlations between different features, while the Deep Network leverages the power of deep neural networks to learn hierarchical representations of the input data.

The core idea of cross network is to apply explicit feature intersections in an efficient way. A cross network consists of intersecting layers, each with the following formula:

= (4)

A few parameters of the cross-network limit the model capacity. To capture highly nonlinear interactions, the model introduces a deep network in parallel.

A deep network is a fully connected feedforward neural network, and each depth layer has the following formula:

(5)

Finally, the top part of DCN is the "Combination Output Layer", which combines the outputs from the Cross Network and Deep Network to produce the final prediction result. This layer plays a crucial role in integrating the complementary strengths of the two networks and improving the overall performance of the model.

The link layer connects the outputs of the two parallel networks and passes through a full link layer to get the output:

(6)

The explanation of DCN v2 model calculation principle.

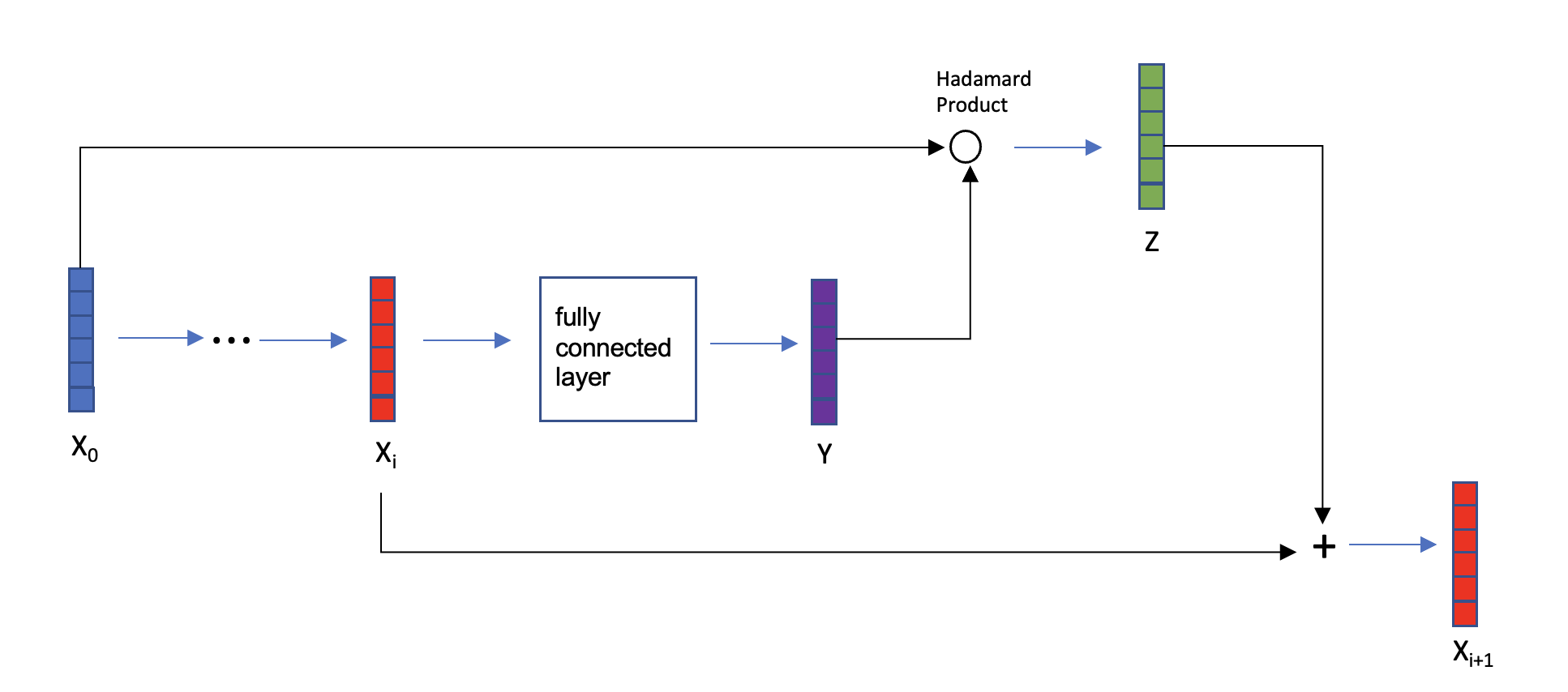


Figure2: Cross-layer

The cross-layer is the fundamental building block of a cross-network, and the figure1 is the structure of a single cross-layer. Given an input vector 0 that passes through neural network layers, the output vector is i. Taking the i-th cross layer as an example, the vector i is fed into a fully connected layer, which generates another vector Y. The lowest-level vector 0 is then subjected to Hadamard product with vector Y to produce the output vector Z. Vectors i and Z represent input and output, respectively, and the sum of these two vectors yields i+1 (similarly to the skip connections in ResNet). Vector i+1 represents the output of the i-th cross layer, while vectors 0 and i are inputs to this layer, and all parameters are contained within the fully connected layer.

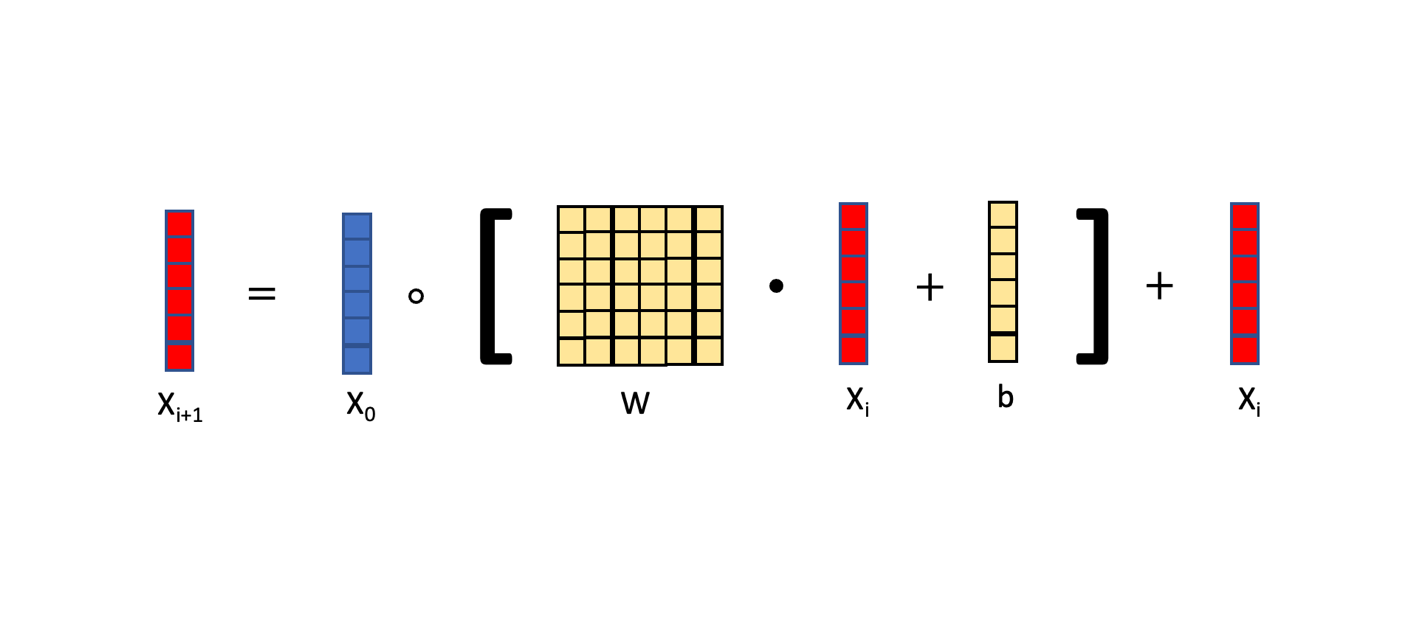


Figure3: formula of Cross-layer

The cross-layer can be represented by the equation in figure 3. The input of the cross-layer consists of two vectors, 0 and i. Here, 0 represents the input at the lowest layer of the neural network, while i represents the input to the i-th layer of the neural network. The square bracketed section denotes a fully connected layer, which computes the product of matrix W and vector i and adds vector b. The output of the fully connected layer is a vector with the same size as the input vector . The matrix W and the vector b are the parameters of the fully connected layer, which need to be updated during training using gradient descent. Finally, the element-wise multiplication (Hadamard product) between the vector 0 and the output of the fully connected layer is taken, followed by addition with vector i. The resulting vector is denoted as i+1, which serves as the output of the cross-layer. Notably, both the input and output are vectors with the same shape.

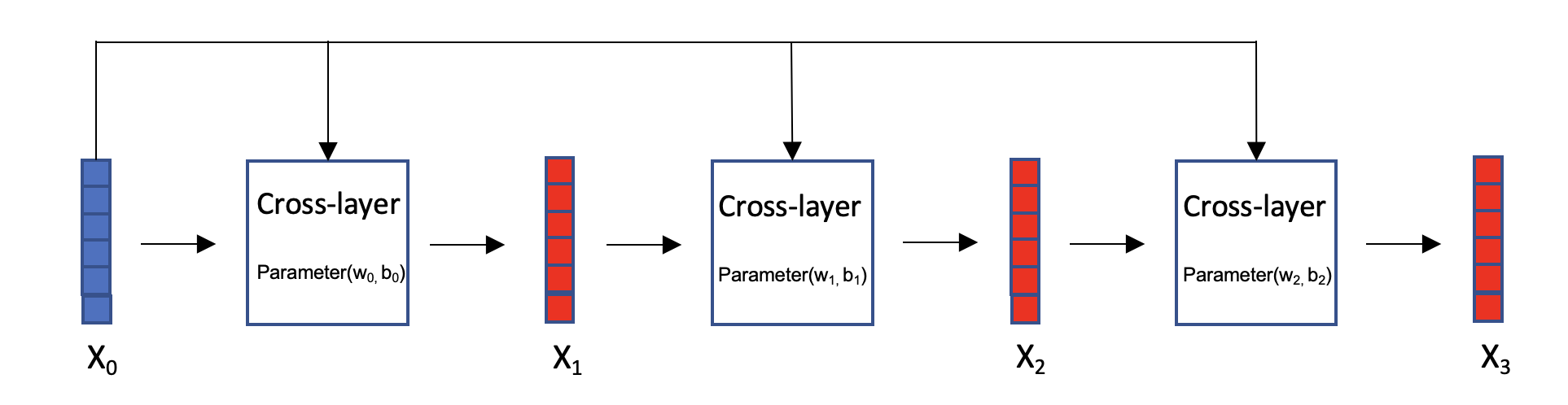


Figure4: Cross network

Cross network, as figure 4: Vector 0 is the input of the cross-network, which is fed into a cross-layer with parameters W0 and b0, resulting in output vector 1. Then, 1 is fed into the next cross-layer, along with 0, both of which serve as inputs to this cross-layer. The parameters for this cross-layer are W1 and b1, and the resulting output vector is 2. This process is repeated iteratively.

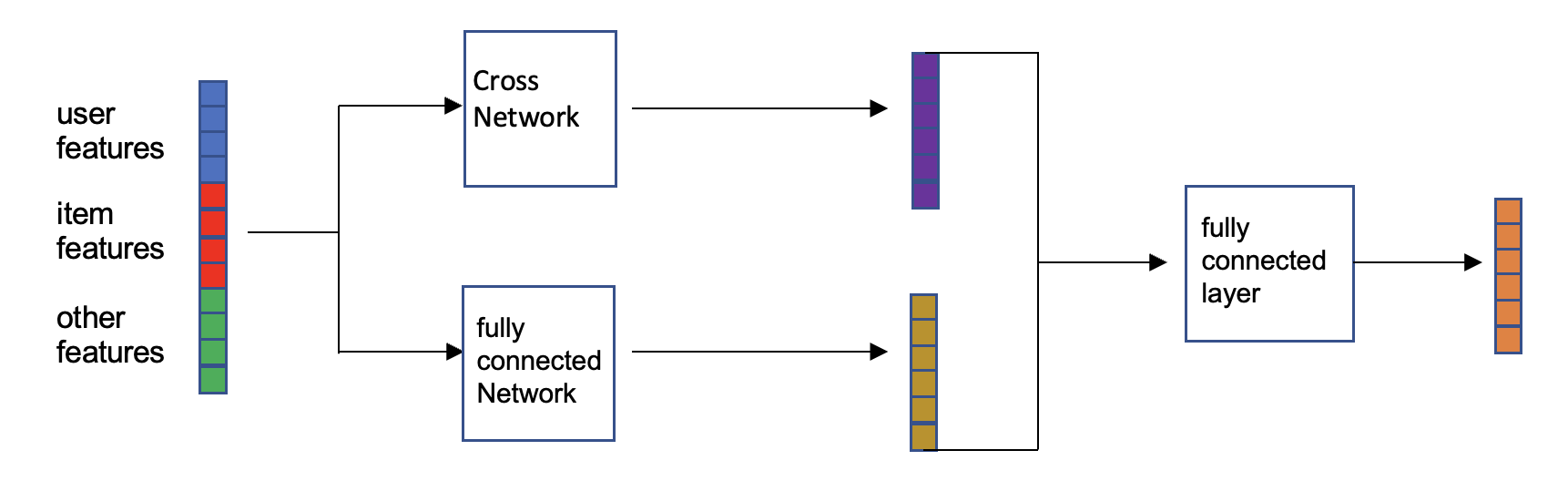


Figure5: Deep & Cross Network

Deep cross network (DCN) is a combination of cross network and fully connected network. In recommendation systems with ranking models, the input includes user features, item features, and other features, which are concatenated and fed into two neural networks in parallel: a fully connected network and a cross-network. Each neural network outputs a vector, which is then concatenated and fed into a fully connected layer that outputs a vector. The concatenation of the fully connected network, cross-network, and fully connected layer is the DCN. The DCN can be used for both recall and ranking.

## **Technology**

These are the hardware and software that I'm going to use in the project

|  |  |
| --- | --- |
| hardware | software |
| Macbook pro 16 INCHES  SoC: m1 pro  gpu: a5000(rent from autodl) | Colab  Pytorch environment  Google Drive  Language: python  Jupyter notebook  Autodl |

Table1：tools

## **Project Version Management**

I use both local storage and Google Cloud to manage my projects, and the papers and data are stored locally before being uploaded to a folder in the cloud and synced. When the codes have been done. It will automatically upload the code to Google Cloud after moving it to the cloud folders. Here is the link: https://drive.google.com/drive/folders/1-1QdZNElGqMBr4bonCHi\_IzAgdo7kKHM?usp=sharing

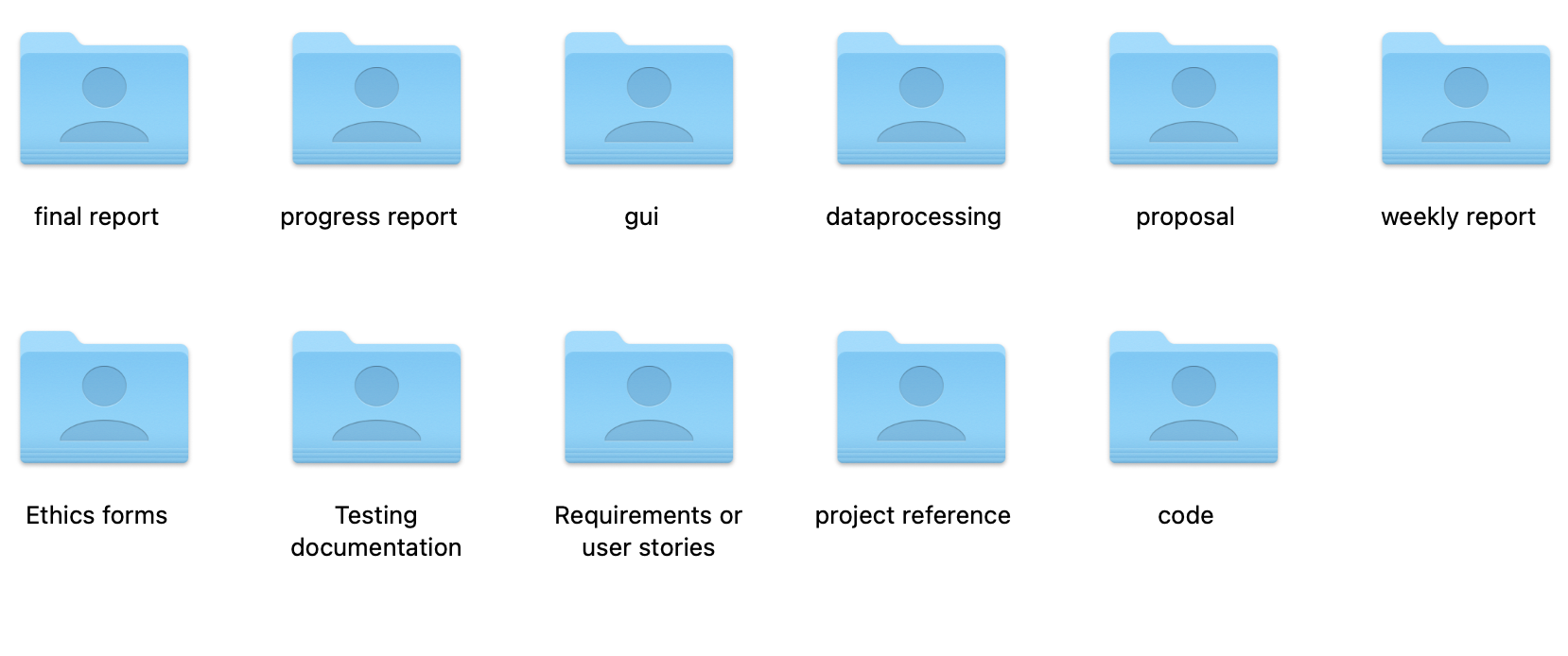


Figure6: management folders

# **Results**

Here students are to provide detailed descriptions and documentation of results and testing. Critical evaluation and discussion of results, issues encountered constraints, limitations, and originality.

The subsection layouts of this section mostly depends on the type of project that the student is carrying out. Students can introduce subsections that will help the readability of their work.

For instance, students doing software development-based projects should provide the detailed use of their software in this chapter. Screenshots (images) of their graphical user interfaces can be depicted in this chapter. Other relevant details about the testing and evaluation of their software can be stated here as well.

Also, students doing research-oriented projects (machine learning & deep learning projects) should state the results of their model training, validation and testing. Use appropriate graphs and figures to illustrate your results. Results from case studies and ablation study of hyper-parameters should be stated here. In a situation where a machine learning-based project was deployed as a web or mobile application, students are to provide details of functionality tests.

# **Professional Issues**

## **Project Management**

### **Activities**

State the complete tasks for each objective. The details here can be presented by a table.

### **Schedule**

In this section, you can use a Gantt chart or other charts to show the activities and their deadlines. Highlight all completed tasks in the project schedule chart.

### **Project Data Management**

In this section, students must describe how they have used resources such as Baidu drive, Gitee, etc., to manage project logs, reports, literature, etc.

### **Project Deliverables**

In this section, briefly list all the documents and project resources that have been submitted for assessment. Example: Project proposal, progress report, final report, project code/ software, poster presentation file, etc.

## **Risk Analysis**

Risk analysis as informed by the current project progress; Resolved risks and the success of the mitigation strategy; Changes to the project plan as a result of risks; Future risks.

## **Professional Issues**

Identification and discussion of relevant legal, social, ethical, and environmental issues in the context of the project. Refer to professional codes of conduct, e.g. BCS, ACM.

# **Conclusion**

Summary of what was achieved and potential future work.

# **References**

* The layout above is a suggestion of how to present your Final Project Report. Whenever appropriate, introduce sections that will help the readability of your work.
* The Length of the final report should be **8000 – 10000 words**.
* All sections and subsections should be numbered for cross-referencing purposes.
* Regarding citations and references, students must adhere to the University guidelines or IEEE referencing style. **Students doing software development-based projects can cite related websites, web applications, developer documentation, etc. They can cite related articles to their projects, but it is not required. Students doing research-oriented projects should focus on citing research articles. They can also cite appropriate websites whenever necessary. Students are advised to use appropriate reference management software such as Mendeley Reference Manager or Zotero to ensure the correctness of all references.**

## **Formatting Requirements**

Your written report must be presented in the following format:

* All main sections/chapters should begin on a new page. The Declaration page, Tables of Contents pages, Acknowledgment, Abstract, Abbreviation, Glossary, Project Chapters (Chapters 1 to 6), and Appendices should all start on a new page.
* It must be word-processed in 11-point Arial font.
* It must be black text on a white or ivory background
* All pages must be numbered. Follow the appropriate page numbering format specified in the template.
* Margins must be as follows: Top: 1 inch, Bottom: 1 inch (2.5 cm), Left: 1.25 inches, Right:
* 1.25 inches (3.2 cm)
* Use a line spacing of 1.5
* Numbers and captions to figures and tables should be at the bottom of the figure or table. If the figure or table is mounted sideways into the report, then its bottom is on the right-hand side of the report. **All tables and figures must be labeled**.
* Normally, the report should not contain more than 80 tables/figures.

## **Written Presentation**

* The final project report must have a concise written presentation and referencing style.
* It should also have a clear & logical presentation.

**NOTE:**

1. **All the text in red colour are basic guidelines and must be DELETED after using this guide.**
2. **Finally, update the “Table of Contents” appropriately to display the correct section titles and corresponding page numbers.**

# **Appendices**

This section can have the essential information/data that are necessary to be included within the report but would disrupt the flow of the main argument. This section is not marked. Examples include links to data and software repositories, questionnaires, raw survey results, and wireframes.