

# 單元 16：簡報編修系統——PDF 轉 PowerPoint

---

前面的單元介紹了兩種自動化工具：Chrome 擴充程式專注於「快速加入來源」，Web 系統專注於「完整操控 NotebookLM」。

這個單元要介紹的是另一個專門化的工具：**簡報編修系統**。

這個系統專注於解決一個特定的問題：把 NotebookLM 生成的 PDF 簡報，轉換成真正可編輯的 PowerPoint 簡報。

## 這個系統要解決什麼問題？

---

讓我們回顧一下 NotebookLM 簡報的編輯困境。

NotebookLM 生成的簡報是 PDF 格式。而這個 PDF，本質上是「圖片的集合」——每一頁都是一張完整的圖片，上面的文字、圖表、裝飾元素全都「壓平」在一起，無法分開編輯。

在前面的單元中，我們介紹了四種後製工具（PDNob、NotebookLM 編輯器、Canva Pro、Lovart），它們各有專精：

- PDNob：用 OCR 技術識別文字，讓你能修改內容
- NotebookLM 編輯器：移除浮水印，轉換成基本的 PPTX
- Canva Pro：用魔法抓取分離元素，提升視覺效果
- Lovart：用 AI 圖層拆解技術，進行深度編輯

這個「簡報編修系統」則採用了不同的策略：**AI 文字移除 + 背景修復 + 重建文字圖層**。

它的目標是生成一份「乾淨」的 PowerPoint：

- 底層是去除文字後的純背景圖片
- 上層是可編輯的文字方塊，位置與原始簡報一致

這樣的結構讓你可以：

- 自由修改任何文字內容

- 保留原始的視覺設計風格
- 調整文字的字體、大小、顏色
- 移動文字到不同的位置

## 系統架構

---

這是一個前後端分離的 Web 應用程式：

### 後端：Flask + Gemini AI

後端使用 Python 的 Flask 框架，負責：

- 接收前端上傳的 PDF 或圖片
- 呼叫 Gemini AI 進行文字移除和背景修復
- 使用 OCR 提取文字內容和位置
- 回傳處理結果

核心的 AI 處理使用 Google 的 Gemini 模型：

- 圖片編輯（文字移除）：gemini-2.5-flash-preview
- 文字生成（OCR）：gemini-2.5-flash

### 前端：Vue 3 + Pinia

前端使用 Vue 3 框架，負責：

- 提供使用者介面
- 處理檔案上傳和預覽
- 顯示處理進度
- 生成最終的 PPTX 檔案

前端使用了幾個重要的函式庫：

- pdfjs-dist：解析 PDF 檔案，將每一頁轉換成圖片
- pptxgenjs：在瀏覽器中生成 PowerPoint 檔案

## 整體流程



前端將新 PDF 拆成每一頁為圖片

↓

使用者選擇要處理的頁面

↓

前端將圖片送到後端

↓

後端呼叫 Gemini AI

1. 提取圖片中的文字
2. 修復錯誤 (inpainting)
3. OCR 提取文字內容和位置

↓

後端回傳：文字圖片 + 文字資訊

↓

前端結合成 PDF

提醒：去字體的頁面圖

提醒：可編輯的文字位置

↓

使用者下載 PDF

## 安裝步驟詳解

這個系統需要同時執行前端和後端服務，安裝步驟比較多，但設計者提供了一鍵啟動的方式。

### 系統需求

項目	需求
Python	3.10 以上
Node.js	18.0 以上
記憶體	8GB RAM 以上
硬碟空間	500MB 以上
網路	需要連網（呼叫 Gemini API）

### 步驟一：安裝 Python

如果你還沒有安裝 Python：

1. 前往 <https://www.python.org/downloads/>
2. 下載 Python 3.10 或更新版本
3. 安裝時**務必勾選**「Add Python to PATH」

4. 驗證安裝：在命令提示字元輸入 `python --version`

## 步驟二：安裝 Node.js

如果你還沒有安裝 Node.js：

1. 前往 <https://nodejs.org/>
2. 下載 LTS 版本（建議 18.x 或 20.x）
3. 執行安裝程式，使用預設選項
4. 驗證安裝：在命令提示字元輸入 `node --version`

## 步驟三：取得 Gemini API Key

這個系統使用 Google 的 Gemini AI 來處理圖片，你需要一個 API Key。

1. 前往 Google AI Studio：<https://aistudio.google.com/apikey>
2. 登入你的 Google 帳號
3. 點擊「Create API key」
4. 選擇專案或建立新專案
5. 複製生成的 API Key（妥善保存，等下會用到）

## 步驟四：一鍵啟動（Windows）

最簡單的啟動方式是使用提供的批次檔。

在檔案總管中，找到並雙擊：

```
start cmd /k cd %~dp0 & python app.py
```

首次執行時，系統會自動安裝所需的套件（Python 和 Node.js 的依賴），這可能需要 1-2 分鐘。

安裝完成後，會自動開啟兩個命令視窗：

- 一個是後端服務（port 8099）
- 一個是前端服務（port 5173）

## 步驟五：開啟系統

在瀏覽器中開啟：

https://www.gemini.com

你會看到一個啟動畫面，幾秒後進入主介面。

## 步驟六：設定 API Key

首次使用時，需要設定 Gemini API Key。

1. 點擊介面右下角的「齒輪」圖示，進入設定頁面
2. 找到 **gemini** 服務商，點擊「編輯」
3. 貼上你在步驟三取得的 API Key
4. 點擊「儲存」

設定完成後，系統就可以正常使用了。

## 使用流程

讓我們走過完整的使用流程。

### 流程概覽

上傳檔案 → 上傳檔案 → 選擇頁面 → AI 處理 → 匯出 PPTX

### 步驟一：上傳檔案

進入系統後，你會看到上傳區域。

支援的格式：

- PDF（最常用）
- JPG、JPEG
- PNG
- WebP
- BMP

你可以：

- 拖放檔案到上傳區域
- 或點擊上傳區域，選擇檔案

如果上傳的是 PDF，系統會自動將每一頁轉換成圖片。

## 步驟二：選擇頁面

上傳完成後，系統會顯示所有頁面的縮圖。

你可以：

- 點擊勾選需要處理的頁面
- 使用「全部選取」一次選擇所有頁面
- 使用「全部取消」清除選擇

**提示：**如果只有少數頁面需要編輯，建議只選擇那些頁面。這樣處理速度更快，也節省 API 用量。

## 步驟三：AI 處理

選擇完頁面後，點擊「開始處理」。

系統會逐頁進行 AI 處理：

1. 移除圖片中的文字
2. 修復被文字遮蓋的背景區域
3. OCR 識別文字內容和位置

每頁大約需要 3-5 秒。處理過程中會顯示即時進度。

## 步驟四：預覽結果

處理完成後，你可以預覽每一頁的結果：

- 查看去除文字後的背景圖
- 確認 OCR 識別的文字是否正確
- 檢查文字位置是否準確

## 步驟五：匯出 PPTX

確認結果無誤後，選擇簡報比例：

- 16:9：橫式寬螢幕，最常用
- 9:16：直式，適合手機或直立螢幕
- 4:3：傳統標準比例

點擊「匯出 PPTX」，下載生成的 PowerPoint 檔案。

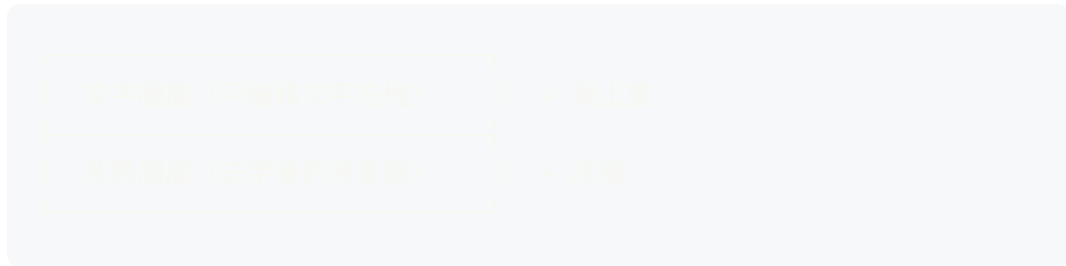
## PPTX 檔案結構

---

理解生成的 PPTX 結構，可以幫助你更有效地編輯。

### 圖層結構

每一頁簡報包含兩個圖層：



**背景圖層：**

- 原始簡報的視覺設計，但文字已被移除
- AI 自動修復了原本被文字遮蓋的區域
- 這一層通常不需要編輯

**文字圖層：**

- OCR 識別出的文字，放在獨立的文字方塊中
- 位置盡量與原始位置一致
- 可以自由編輯、移動、刪除、新增

### 編輯技巧

打開生成的 PPTX 後，你可以：

**修改文字內容：** 直接點擊文字方塊，修改裡面的內容。

**調整文字樣式：** 選取文字後，可以改變字體、大小、顏色、粗細等。

**移動文字位置：** 拖曳文字方塊到新的位置。

**刪除不需要的文字：** 選取文字方塊，按 Delete 刪除。

**新增文字：** 使用 PowerPoint 的「插入文字方塊」功能，新增額外的文字。

**調整背景：** 如果需要調整背景圖片，可以點擊背景圖層進行編輯。但要注意不要遮住了文字。

## 處理品質說明

---

AI 處理的品質會受到幾個因素影響。

### 文字移除品質

**效果較好的情況：**

- 背景較為簡單（純色、漸層）
- 文字與背景對比明顯
- 文字區域沒有太複雜的圖形

**效果可能較差的情況：**

- 背景非常複雜（如照片）
- 文字壓在圖表或插圖上
- 文字使用了透明或漸層效果

AI 修復背景時，會根據周圍的視覺資訊「猜測」被遮蓋區域的內容。如果周圍的資訊太複雜，修復結果可能不夠完美。

### OCR 準確度

**效果較好的情況：**

- 文字清晰、字體標準
- 原始 PDF 解析度高
- 使用常見的字體

**效果可能較差的情況：**

- 文字太小或太模糊
- 使用裝飾性字體
- 文字有旋轉或變形

**關於位置準確度：**

- **PDF 來源：** 使用原生的文字座標，位置非常準確



- **圖片來源：**使用 AI 推測的位置，可能有些微偏差

建議：盡量使用 PDF 格式作為輸入，這樣文字位置會更準確。如果使用圖片，匯出後可能需要在 PowerPoint 中微調文字位置。

## 與其他工具的比較

讓我們比較這個系統和前面介紹的其他工具。

### 與 NotebookLM 編輯器比較

比較項目	NotebookLM 編輯器	簡報編修系統
核心技術	OCR + 浮水印移除	AI 文字移除 + 背景修復
背景處理	保留原始背景（含文字）	移除文字並修復背景
文字位置	OCR 推測位置	PDF 原生座標 + OCR
執行方式	雲端服務	本機執行
費用	免費額度有限	需 Gemini API（按量計費）
適合情況	快速轉換	需要乾淨背景的情況

簡單來說：

- NotebookLM 編輯器適合「快速轉換」——轉成 PPTX 後再編輯
- 簡報編修系統適合「乾淨背景」——需要完全重建文字圖層的情況

### 與 Canva Pro 比較

比較項目	Canva Pro	簡報編修系統
核心技術	魔法抓取（元素分離）	AI 文字移除 + 重建
操作方式	手動分離、調整	全自動處理
彈性	高（可任意調整）	低（批量處理）
適合情況	視覺升級、設計調整	大量簡報轉換

簡單來說：

- Canva Pro 適合「精細調整」——一頁一頁地優化視覺效果
- 簡報編修系統適合「批量轉換」——快速處理多頁簡報

## 進階設定

---

### 選擇不同的 AI 服務商

這個系統預設使用 Google Gemini，但架構上支援更換為其他 AI 服務商。

在 `providers.yaml` 設定檔中，可以配置：

```
active_provider: gemini # 目前使用的服務商

providers:
  gemini:
    type: gemini
    api_key: "YOUR_API_KEY_HERE"
    model_image_edit: gemini-2.5-flash-preview-05-20
    model_text_gen: gemini-2.5-flash
```

如果你有其他 AI 服務商的帳號（如 OpenAI），可以在這裡新增配置。

### 調整模型

預設使用的模型是：

- 圖片編輯： `gemini-2.5-flash-preview-05-20`
- 文字生成： `gemini-2.5-flash`

如果你想使用其他模型（例如更強大但更貴的版本），可以在設定檔中修改。

## 常見問題與解決方案

---

### 問題一：啟動後顯示「未配置有效的 API Key」

原因：還沒有設定 Gemini API Key。

解決方法：

1. 確認已取得 Gemini API Key
2. 進入設定頁面
3. 找到 gemini 服務商，貼上 API Key

#### 4. 儲存後重新整理頁面

### 問題二：Python 或 Node.js 找不到

原因：安裝時沒有加入系統 PATH，或需要重新開啟終端機。

解決方法：

1. 確認安裝 Python 時有勾選「Add to PATH」
2. 確認安裝 Node.js 時使用預設選項
3. 重新開啟命令提示字元視窗
4. 執行 `python --version` 和 `node --version` 確認

### 問題三：處理速度很慢

可能原因：

1. 網路連接不穩定
2. Gemini API 速率限制
3. 圖片解析度過高

解決方法：

1. 檢查網路連接
2. 減少同時處理的頁面數量
3. 如果持續緩慢，等待幾分鐘後重試

### 問題四：匯出的 PPTX 文字位置不準確

說明：

- 如果來源是 PDF，文字位置應該很準確
- 如果來源是圖片，OCR 推測的位置可能有偏差

建議：

- 盡量使用 PDF 格式作為輸入
- 匯出後在 PowerPoint 中微調文字位置

### 問題五：某些文字沒有被移除乾淨

可能原因：

- 文字與背景的對比不夠明顯
- 文字使用了特殊效果（陰影、外框、漸層等）
- 背景過於複雜

**解決方法：**

- 這種情況下，可能需要在 PowerPoint 中手動處理殘留的文字
- 或者考慮使用 Lovart 進行更精細的圖層編輯

## 使用情境範例

---

### 情境一：教學講義轉換

王老師用 NotebookLM 生成了一系列的教學講義簡報，共 50 頁。他想要在 PowerPoint 中進行一些修改，包括：

- 更新某些頁面的數據
- 調整部分文字的說明
- 在某些頁面新增補充內容

**傳統方式：**用 PDNot 或 NotebookLM 編輯器處理，但 50 頁逐頁編輯太耗時。

**使用簡報編修系統：**

1. 上傳 50 頁的 PDF
2. 全選所有頁面
3. 開始批量處理（約 3-4 分鐘）
4. 匯出 PPTX
5. 在 PowerPoint 中進行修改

整個轉換過程只需要幾分鐘，之後就可以在 PowerPoint 中自由編輯所有內容。

### 情境二：品牌化簡報

某公司的行銷部門使用 NotebookLM 生成了產品介紹簡報，但需要加入公司的品牌元素：

- 統一的標題字體

- 品牌色系的文字顏色
- 頁首頁尾的公司 Logo

**使用簡報編修系統：**

1. 上傳 NotebookLM 生成的 PDF
2. 處理並匯出 PPTX
3. 在 PowerPoint 中：
  - 修改所有標題的字體
  - 調整文字顏色為品牌色
  - 使用「投影片母片」功能加入 Logo

因為文字都在獨立的文字方塊中，這些調整變得非常容易。

### 情境三：多語言版本

一家跨國公司需要把同一份簡報翻譯成多種語言。

**使用簡報編修系統：**

1. 上傳中文版的 NotebookLM 簡報
2. 處理並匯出 PPTX
3. 在 PowerPoint 中直接修改文字為其他語言
4. 儲存為不同語言的版本

因為背景和文字是分開的，翻譯時只需要修改文字圖層，視覺設計完全保留。

## 本章小結

---

簡報編修系統是一個專門化的工具，專注於解決「PDF 簡報轉換為可編輯 PPTX」的問題。它的核心價值是：

**AI 文字移除：**使用 Gemini AI 移除圖片中的文字

**背景修復：**自動修復被文字遮蓋的區域

**重建文字圖層：**將文字放在獨立的文字方塊中，可自由編輯

**批量處理：**可以一次處理多頁簡報

這個系統特別適合需要大量編輯 NotebookLM 簡報的情況。轉換完成後，你就擁有了一份「真正可編輯」的 PowerPoint——不是「看起來像 PowerPoint 的圖片集」，而是可以自由修改每一個文字的專業簡報。

下一個單元，我們會介紹這個系統的部署方式，包括如何讓它在區域網路內分享使用，以及如何設定遠端存取。