

Contents

1 程新伟专属转换器——PDF 转 Word 神器 1

1.1 功能亮点 1

1.2 界面预览 2

1.3 快速开始 2

1.3.1 方式一：直接用打包好的 exe（推荐懒人使用） 2

1.3.2 方式二：从源码运行（适合爱折腾的你） 2

1.4 一键打包 3

1.5 项目结构 3

1.6 技术栈 4

1.7 设置说明 4

1.8 常见问题 4

1.9 依赖清单 5

1.10 更新日志 5

1.10.1 v1.2 (2026-02-10) 5

1.11 许可证 5

1 程新伟专属转换器——PDF 转 Word 神器

一个长得好看、用着顺手、转得飞快的 PDF 转 Word 工具。
专治各种”PDF 改不了”的疑难杂症，从此告别手动抄文档的苦日子！

Python Platform License GUI Status

1.1 功能亮点

功能	说明	状态
PDF → Word	高质量转换，保留排版格式	
实时进度条	显示百分比、当前页码、预计剩余时间	
页范围选择	只转你想转的那几页，不浪费时间	
自定义背景	换张好看的壁纸，心情都好了	
透明面板	背景图透出来，颜值拉满	
自定义标题	想叫啥名就叫啥名	
超时提示	遇到复杂页会贴心提醒”别急，在努力”	
异常页跳过	个别烂页不影响整体，转完告诉你跳了哪些	
设置持久化	关掉再开，背景和标题都还在	

1.2 界面预览

程新伟专属转换器	
PDF转Word工具（支持大文件）	
选择PDF文件	
<input type="text" value="C:\xxx\论文.pdf"/>	<input type="button" value="浏览..."/>
页范围（可选）	
起始页：[] 结束页：[]	
<div><div></div>72%</div>	
生成页 18/25 （72%）	
<input type="button" value="开始转换"/>	<input type="button" value="清除"/>
就绪	

1.3 快速开始

1.3.1 方式一：直接用打包好的 exe（推荐懒人使用）

双击 `dist/PDF 转换工具.exe`，开箱即用，不需要装 Python，不需要装任何东西。

首次启动可能稍慢（几秒），别急，它在解压自己，又不是卡了。

1.3.2 方式二：从源码运行（适合爱折腾的你）

1.3.2.1 1 克隆项目

```
git clone https://github.com/yourname/PDFtoWORD.git
cd PDFtoWORD
```

1.3.2.2 2 创建虚拟环境 (推荐 Python 3.8, 兼容 Win7)

```
py -3.8 -m venv .venv38
.venv38\Scripts\activate
```

1.3.2.3 3 安装依赖

```
pip install -r requirements.txt
```

1.3.2.4 4 启动

```
python pdf_converter.py
```

1.4 一键打包

想把它打包成一个独立的 exe 分发给朋友？一条命令搞定：

```
python -m PyInstaller --onefile --windowed --name "PDF 转换工具" --clean pdf_converter.py
```

参数	含义
<code>--onefile</code>	打包成单个 exe 文件，方便分发
<code>--windowed</code>	不弹出黑色控制台窗口
<code>--name</code>	给 exe 起个好听的名字
<code>--clean</code>	打包前清理缓存，确保干净

打包完成后，exe 在 `dist/PDF 转换工具.exe`。

兼容性提示：用 Python 3.8 打包可兼容 Win7/10/11；用 Python 3.9+ 打包则只支持 Win10+。

1.5 项目结构

PDFtoWORD/

```

├─ pdf_converter.py      # 主程序（所有代码都在这，简单粗暴）
├─ requirements.txt      # 依赖清单
├─ README.md            # 你正在看的这个文件
├─ 更新说明.md          # 版本更新记录
├─ 打包说明.md          # 打包相关说明
├─ create_test_pdf.py   # 测试 PDF 生成器
├─ dist/                # 打包输出目录
├   └─ PDF转换工具.exe   # 打包好的可执行文件
├─ build/               # PyInstaller 构建临时文件
└─ .venv38/             # Python 3.8 虚拟环境

```

1.6 技术栈

技术	用途	为什么选它
Python 3.8	主语言	支持 Win7，够稳
tkinter	GUI 框架	Python 自带，不用额外装
pdf2docx	PDF 转 Word 核心引擎	转换质量高，社区活跃
Pillow	图片处理	背景图缩放、透明混合
PyInstaller	打包工具	一键生成 exe

1.7 设置说明

点击界面左上角的 **** 齿轮按钮，打开设置窗口：

- **** 标题文字 ****：修改主界面大标题，想叫”摸鱼转换器”也行
- **** 更换背景 ****：选一张喜欢的图片作为背景（自动保存到程序目录）
- **** 面板透明度 ****：滑块调节 20%~100%，越低越透明，越能看到背景图

所有设置保存在程序目录下的 `settings.json`，删掉它就恢复默认。

1.8 常见问题

Q：转换出来格式不对怎么办？

A：PDF 转 Word 本身就是个”尽力而为”的事情。简单排版的文档效果很好，复杂的表格嵌套、花式排版可能会有偏差。建议转完后手动微调一下。

Q: 转换大文件好慢啊!

A: 259 页的论文确实需要一些时间, 请耐心等待。进度条会告诉你预计剩余时间。如果某一页特别慢 (超过 60 秒), 状态栏会提示”该页复杂请耐心等待”。

Q: exe 在 Win7 上报错 api-ms-win-core-path 缺失?

A: 你用了 Python 3.9+ 打包。请用 Python 3.8 的虚拟环境重新打包, 详见上方打包说明。

Q: 为什么 exe 这么大 (100MB+)?

A: 因为它把 Python 解释器、numpy、opencv 等一堆依赖全塞进去了。功能多, 体积大, 能理解的对吧

Q: 能转 Excel 吗?

A: 不能。这是一个专注于 PDF→Word 的工具, 专注做一件事, 做到最好

1.9 依赖清单

pdf2docx # PDF 转 Word 核心库

Pillow # 图片处理 (背景图功能)

打包时还需要:

pyinstaller # exe 打包工具

1.10 更新日志

1.10.1 v1.2 (2026-02-10)

- 全新 Canvas 透明面板, 背景图不再被遮挡
 - 支持自定义背景图片和透明度调节
 - 支持自定义标题文字
 - 实时进度条: 百分比 + 页码 + 预计剩余时间
 - 单页超时提醒 (60 秒阈值)
 - 支持指定页范围转换
 - 异常页自动跳过并汇总报告
 - 使用 Python 3.8 打包, 兼容 Win7/10/11
-

1.11 许可证

本项目采用 MIT 许可证, 随使用, 随便改, 记得请作者喝杯咖啡就好

用和驱动开发如果觉得好用，给个呗 ~