1.摘出标题，图标和摘要

グラフィカル ユーザー インターフェイス

自動的に生成された説明

标题，摘要和图表

正则表达式 图标的话用python自带的Pdf提取软件，或者使用OCR，各有各的好坏

Python的库：可以识别图片，但是某些多个小图拼凑起来的图片，效果会非常不好。如图1所示，这是原论文的图，但是提取出来的效果如图2所示，效果非常的差。但是可以较好的将文字区别出来。

グラフィカル ユーザー インターフェイス, アプリケーション

自動的に生成された説明

图1 效果不好的例子

Word

低い精度で自動的に生成された説明

图2 提取出的效果

第二种方案是用OCR，对于图片里有文字的话，也会将文字进行扫描。

グラフィカル ユーザー インターフェイス, アプリケーション

自動的に生成された説明

グラフィカル ユーザー インターフェイス, テキスト, アプリケーション

自動的に生成された説明

2.文本推荐

グラフィカル ユーザー インターフェイス, アプリケーション

自動的に生成された説明

系统推荐论文：bert/doc2vec相似度

グラフィカル ユーザー インターフェイス, アプリケーション

自動的に生成された説明

由于不能储存，应该是计算每篇论文的文章向量，用Cosine Similarity进行相似度的比较

グラフィカル ユーザー インターフェイス, アプリケーション, メール

自動的に生成された説明

这一步的话，已经有原论文的abstract和summary了，为什么需要简化。直接提取abstract和summary效果估计会更好

グラフィカル ユーザー インターフェイス, テキスト, アプリケーション

自動的に生成された説明

使用transformer或者其他SOTA模型，或者调用谷歌翻译api？