Projet de Développement

INF1603 réalisé

28 janvier 2021

Table des matières

Sprint 1 - Rapport

28 janvier 2021

Résumé

L'INRIA, par manque de temps, souhaite avoir un outil capable de résumer des documents au format PDF pour ensuite être traités par un système TAL (*Traitement Automatique de Langues*), en format texte. Cet outil doit, non seulement fonctionner sur des systèmes GNU/Linux, mais aussi être capable de bien découper les sections d'un article, tout en prenant en compte la gestion des textes en double colonnes, formules mathématiques, figures etc.

Deux outils libres sont à notre disposition pour la conversion de PDF vers TXT : `pdftotext` et `pdf2txt`.

Notre premier objectif est donc de comparer les deux outils pour déterminer lequel des deux est le plus apte à répondre aux contraintes demandées. Cette étude mettra en exergue, par une analyse comparative, certaines limites, mais aussi les points forts de ces deux commandes.

1 Commandes

La première commande que nous avons utilisée est :

```
pdftotext input.pdf output.txt
```

Cette commande sans option convertit un fichier PDF (Portable Document Format) en texte brut dans un fichier (output.txt). Le fichier est lu de gauche à droite, de haut en bas et la sortie est affichée dans le même ordre. Les sauts de lignes sont conservés. Pas de double

colonnes. La première phrase (en terme de hauteur) sera lue et écrite en premier dans le fichier de sortie.

La première option que nous avons utilisée est :

```
pdftotext -layout input.pdf output.txt
```

Celle-ci permet de conserver la mise en page initiale du fichier PDF. Cela permet, en effet, de conserver les espaces, tabulations, et la mise en forme du texte. Par exemple, s'il y avait initialement un texte sur 2 colonnes, la disposition en 2 colonnes sera gardée.

La deuxième option que nous avons utilisée est :

```
pdftotext -raw input.pdf output.txt
```

Celle-ci est identique à la commande sans option, mais permet de supprimer les sauts de lignes, espaces abusifs.

Nous n'avons pas utilisé les autres options, car elles n'apportaient pas de différence au fichier de sortie. Par exemple, l'une d'entre elles aurait permit de choisir la page à partir de laquelle commencer à extraire le texte. Ce qui n'est pas demandé.

Ensuite, la deuxième commande dont nous nous sommes servis est :

```
pdf2txt input.pdf -t text -o output.txt
```

L'option `-t text` permet de forcer le bon format de sortie. Elle n'apporte aucun changement au fichier texte. Cette commande lit le fichier de gauche à droite et de haut en bas. Elle conserve les sauts de lignes. Les textes en double colonnes sont concaténés pour ne former qu'un seul et unique texte sans espace.

L'unique option que nous avons utilisée est :

```
pdf2txt input.pdf -t text -o output.txt -A
```

Celle-ci permet de forcer la disposition du texte, et ainsi permettre une lecture plus aérée, notamment grâce à la conservation des sauts de lignes. Pour les textes en double colonnes,

l'affichage se fait dans le sens de la lecture.

Les deux commandes ne permettent ni l'extraction, ni l'affichage des images. Les deux logiciels sont open-sources.

La différence entre ces deux commandes, est d'une part, que `pdftotext` est un paquet, utilisable sur la plupart des distributions Linux. Ce logiciel est inclus dans le package `poppler-utils`. Et d'autre part, la commande `pdf2txt`, moins complète en options, est totalement faite en python et provient d'une suite de logiciels, eux aussi faits en python, qui sont des outils sur l'analyse des PDF.

2 Comparatif détaillé

Nous avons listé des points essentiels lors du comparatif de ces deux commandes dans le tableau ci-dessous.

Critères de comparaison	**pdftotext**	**-layout**	**-raw**	**pdf2txt**	**pdf2txt -A**
Gestion des pieds de page	non	bien	non	non	bien
Conservation du format de l'article [a]	non	très bien	non	non	non
Prise en compte des doubles colonnes [b]	bien	non	oui	oui	oui
Mots bien découpés ?	très bien	très bien	passable	passable	bien
Compacité du texte	très bien	insuffisant	très bien	trop	très bien
Formules de maths lisibles	insuffisant	bien	passable	non	insuffisant
Affichage lisible des tableaux	passable	très bien	passable	insuffisant	bien

a. Affichage comme le fichier original, sous la forme de 2 colonnes
b. Affichage des deux colonnes, en une seule

3 Conclusion

Pour conclure, nous avons remarqué que, de manière générale, la globalité du texte est bien retransmise, notre étude s'est donc portée sur l'analyse de la conversion des tableaux, des doubles colonnes, des pieds de page, des images, ou encore des formules mathématiques. D'après le tableau représenté ci-dessus en page 5, nous avons remarqué qu'aucune des options, ni de `pdftotxt`, ni de `pdf2text` ne convertit en prenant en compte l'ensemble de nos critères. Nous en déduisons donc, qu'il n'y a pas un utilitaire plus performant qu'un autre, mais que nous devons choisir ce que nous allons utiliser en fonction de nos besoins et attentes.

Tout d'abord, si l'on veut retrouver la mise en page originale du document, `pdftotext` avec l'option `-layout` sera la plus adaptée, notamment pour une double colonne, mais aussi, pour permettre la lecture des tableaux, voire des formules quand elles ne comportent pas de symboles spéciaux, comme $\sum$ ou β par exemple. Cette option prend aussi en compte les pieds de page, ce qui peut être intéressant pour retrouver un certain passage dans le document originel.

Ensuite, si l'on recherche un format de sortie pour y appliquer un *Traitement Automatique de Langues* (TAL), une sortie sur une colonne est à privilégier et il vaut mieux choisir l'utilitaire `pdftotext` cette fois-ci encore, mais sans option. En effet, l'option `-raw` ne rend qu'un seul et unique bloc de texte (en fonction de l'ordonnancement du texte dans le fichier original), sans démarcations des paragraphes ce qui ne facilite pas sa lecture.

Cependant, cette méthode ne traite pas bien la démarcation des titres, ce que fait en revanche l'utilitaire `pdf2txt` avec l'option `-A`. A l'inverse, il ne faut pas que le PDF d'origine contienne des images, sans quoi l'ordre des paragraphes pourrait s'en retrouver modifié.

Toutefois, nous n'avons pas parlé de l'option `-htmlmeta` pour la commande `pdftotext`, celle-ci permet d'avoir un fichier de sortie au format html, ce qui nous permet de bien distinguer le début et la fin de certaines portions du texte original, comme le titre ou les auteurs via des balises, les méta-informations (i.e : `<title>[...]</title>`).

Enfin, avec ces résultats, nous ne pouvons conclure qu'il y a un logiciel plus adapté qu'un autre. Il faut donc prendre en compte vos besoins et les documents donnés, afin d'opter pour la commande la plus avantageuse.