Convert HTML-Strings to LaTex using pypandoc

stackoverflow.com/questions/17030391/convert-html-strings-to-latex-using-pypandoc

I am trying to convert HTML-strings to LaTex using pypandoc (a python wrapper for Pandoc).

Coverting files using pypandoc works just fine:

```
import pypandoc

input = 'SomeFile.html'
output = pypandoc.convert(input, 'tex')
```

But if I try to pass some string (which should be possible according to the pypandoc package index if you define the strings format) i get an `IOError: [Errno 63] File name too long:` :

```
input = '''HTML-string'''
output = pypandoc.convert(input, 'tex', format='html')
```

Somehow a file is expected even if I specify `format='html'` .

I also tried to work around this issue by using the StringIO module, but without success:

```
import pypandoc
import StringIO

output = StringIO.StringIO()
output.write('''HTML-string''')
contents = output.getvalue()
output.close()

convertedOutput = pypandoc.convert(contents, 'tex', format='html')
```

I am new to python and would really appreciate some help or hints. Thanks in advance!

python pandoc

2 Answers

up vote 0 down vote
If you check `pypandoc` sources you see that `convert` just runs `pandoc` process with proper input and output streams.

The error happens when `pandoc` command is not found. Probably you installed `pypandoc` and forget about `pandoc` itself. Or the command is out of your shell `PATH` .

answered Jun 11 '13 at 15:19

Maksym Polshcha
12.3k63762

up vote 0 down vote
In case someone needs the answer, here is a minimal workign example using the `subprocess` module and reading the input from `stdin` and outputting the converted string on `stdout` .

```
# -*- coding: utf8 -*-

import subprocess
import os

PANDOC_PATH = r"path/to/pandoc"

def convert(text_to_convert):

    pandoc = subprocess.Popen([os.path.join(PANDOC_PATH, 'pandoc.exe'), '-f', 'html', '-
t', 'latex'], stdout=subprocess.PIPE, stdin=subprocess.PIPE, stderr=subprocess.PIPE)
    output, error = pandoc.communicate(text_to_convert.encode('utf-8'))
    converted_output = output

    return converted_output.decode()
```

answered Dec 30 '14 at 22:39

user740316