🏠   Modules   Data connection   Document loaders   File Directory

File Directory

This covers how to load all documents in a directory.

Under the hood, by default this uses the UnstructuredLoader

```python
from langchain.document_loaders import DirectoryLoader
```

We can use the `glob` parameter to control which files to load. Note that here it doesn't load the `.rst` file or the `.html` files.

```python
loader = DirectoryLoader('../', glob="**/*.md")
```

```python
docs = loader.load()
```

```python
len(docs)
```

```
    1
```

Show a progress bar

By default a progress bar will not be shown. To show a progress bar, install the `tqdm` library (e.g. `pip install tqdm`), and set the `show_progress` parameter to `True`.

```python
loader = DirectoryLoader('../', glob="**/*.md", show_progress=True)
docs = loader.load()
```

```
    Requirement already satisfied: tqdm in
/Users/jon/.pyenv/versions/3.9.16/envs/microbiome-
app/lib/python3.9/site-packages (4.65.0)
```

```
0it [00:00, ?it/s]
```

Use multithreading

By default the loading happens in one thread. In order to utilize several threads set the `use_multithreading` flag to true.

```
loader = DirectoryLoader('../', glob="**/*.md",
use_multithreading=True)
docs = loader.load()
```

Change loader class

By default this uses the `UnstructuredLoader` class. However, you can change up the type of loader pretty easily.

```
from langchain.document_loaders import TextLoader
```

```
loader = DirectoryLoader('../', glob="**/*.md", loader_cls=TextLoader)
```

```
docs = loader.load()
```

```
len(docs)
```

```
1
```

If you need to load Python source code files, use the `PythonLoader`.

```
from langchain.document_loaders import PythonLoader
```

```python
loader = DirectoryLoader('../../../../../', glob="**/*.py",
loader_cls=PythonLoader)
```

```python
docs = loader.load()
```

```python
len(docs)
```

```
    691
```

Auto detect file encodings with TextLoader

In this example we will see some strategies that can be useful when loading a big list of arbitrary files from a directory using the `TextLoader` class.

First to illustrate the problem, let's try to load multiple text with arbitrary encodings.

```python
path = '../../../../../tests/integration_tests/examples'
loader = DirectoryLoader(path, glob="**/*.txt", loader_cls=TextLoader)
```

A. Default Behavior

```python
loader.load()
```

```html
<pre style="white-space:pre;overflow-x:auto;line-height:normal;font-family
Mono',consolas,'Courier New',monospace"><span style="color: #800000; text-
#800000">──────────────────────────────── </span><span style="color: #800(
weight: bold">Traceback </span><span style="color: #bf7f7f; text-decoratic
(most recent call last)</span><span style="color: #800000; text-decoration
──────────────────────────────</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #bfbf7f">/data/source/langchain/langchain/document_loaders/</span><
decoration-color: #808000; font-weight: bold">text.py</span>:<span style='
#0000ff">29</span> in <span style="color: #00ff00; text-decoration-color:
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
```

```
<span style="color: #800000; text-decoration-color: #800000">│</span>     <s
decoration-color: #7f7f7f">26 </span><span style="color: #7f7f7f; text-ded
</span>text = <span style="color: #808000; text-decoration-color: #808000'
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>     <s
decoration-color: #7f7f7f">27 </span><span style="color: #7f7f7f; text-ded
<span style="color: #0000ff; text-decoration-color: #0000ff">with</span> <
decoration-color: #00ffff">open</span>(<span style="color: #00ffff; text-c
#00ffff">self</span>.file_path, encoding=<span style="color: #00ffff; text
#00ffff">self</span>.encoding) <span style="color: #0000ff; text-decoratic
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>     <s
decoration-color: #7f7f7f">28 </span><span style="color: #7f7f7f; text-ded
</span><span style="color: #0000ff; text-decoration-color: #0000ff">try</s
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #800000">❯ </span>29 <span style="color: #7f7f7f; text-decoration-c
</span>text = f.read()
text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>     <s
decoration-color: #7f7f7f">30 </span><span style="color: #7f7f7f; text-ded
</span><span style="color: #0000ff; text-decoration-color: #0000ff">except
text-decoration-color: #00ffff">UnicodeDecodeError</span> <span style="col
#0000ff">as</span> e:                                    <spa
color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>     <s
decoration-color: #7f7f7f">31 </span><span style="color: #7f7f7f; text-ded
</span><span style="color: #0000ff; text-decoration-color: #0000ff">if</sr
decoration-color: #00ffff">self</span>.autodetect_encoding:
style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>     <s
decoration-color: #7f7f7f">32 </span><span style="color: #7f7f7f; text-ded
</span>detected_encodings = <span style="color: #00ffff; text-decoration-c
#00ffff">self</span>.detect_file_encodings()                 <span s
color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #bfbf7f">/home/spike/.pyenv/versions/3.9.11/lib/python3.9/</span><s
decoration-color: #808000; font-weight: bold">codecs.py</span>:<span style
#0000ff">322</span> in <span style="color: #00ff00; text-decoration-color:
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>     <s
decoration-color: #7f7f7f"> 319 </span><span style="color: #7f7f7f; text-c
<span style="color: #0000ff; text-decoration-color: #0000ff">def</span> <s
```

```
decoration-color: #00ff00">decode</span>(<span style="color: #00ffff; text
<span style="color: #00ffff; text-decoration-color: #00ffff">input</span>,
decoration-color: #0000ff">False</span>):
#800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f"> 320 </span><span style="color: #7f7f7f; text-c
<span style="color: #7f7f7f; text-decoration-color: #7f7f7f"># decode inpu
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f"> 321 </span><span style="color: #7f7f7f; text-c
</span>data = <span style="color: #00ffff; text-decoration-color: #00ffff"
#00ffff; text-decoration-color: #00ffff">input</span>
style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #800000">❱ </span> 322 <span style="color: #7f7f7f; text-decoratior
consumed) = <span style="color: #00ffff; text-decoration-color: #00ffff">s
style="color: #00ffff; text-decoration-color: #00ffff">self</span>.errors,
style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f"> 323 </span><span style="color: #7f7f7f; text-c
<span style="color: #7f7f7f; text-decoration-color: #7f7f7f"># keep undeco
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f"> 324 </span><span style="color: #7f7f7f; text-c
<span style="color: #00ffff; text-decoration-color: #00ffff">self</span>.b
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f"> 325 </span><span style="color: #7f7f7f; text-c
<span style="color: #0000ff; text-decoration-color: #0000ff">return</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color:
#800000">└
<span style="color: #ff0000; text-decoration-color: #ff0000; font-weight:
style="color: #008000; text-decoration-color: #008000">'utf-8'</span> code
#008080; text-decoration-color: #008080; font-weight: bold">0xca</span> in
text-decoration-color: #008080; font-weight: bold">0</span>: invalid conti

<span style="font-style: italic">The above exception was the direct cause

<span style="color: #800000; text-decoration-color: #800000">┌
style="color: #800000; text-decoration-color: #800000; font-weight: bold">
#bf7f7f; text-decoration-color: #bf7f7f; font-weight: bold">(most recent c
#800000; text-decoration-color: #800000"> ─
<span style="color: #800000; text-decoration-color: #800000">│</span> in <
decoration-color: #00ff00">&lt;module&gt;</span>:<span style="color: #0000
#0000ff">1</span>
style="color: #800000; text-decoration-color: #800000">│</span>
```

```
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #800000">❱ </span>1 loader.load()
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">2 </span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #bfbf7f">/data/source/langchain/langchain/document_loaders/</span><
decoration-color: #808000; font-weight: bold">directory.py</span>:<span st
color: #0000ff">84</span> in <span style="color: #00ff00; text-decoration-
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">81 </span><span style="color: #7f7f7f; text-ded
│   </span><span style="color: #0000ff; text-decoration-color: #0000ff">i
text-decoration-color: #00ffff">self</span>.silent_errors:
style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">82 </span><span style="color: #7f7f7f; text-ded
│   │    </span>logger.warning(e)
decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">83 </span><span style="color: #7f7f7f; text-ded
│   </span><span style="color: #0000ff; text-decoration-color: #0000ff">e
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #800000">❱ </span>84 <span style="color: #7f7f7f; text-decoration-
</span><span style="color: #0000ff; text-decoration-color: #0000ff">raise
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">85 </span><span style="color: #7f7f7f; text-ded
</span><span style="color: #0000ff; text-decoration-color: #0000ff">final
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">86 </span><span style="color: #7f7f7f; text-ded
│   </span><span style="color: #0000ff; text-decoration-color: #0000ff">i
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">87 </span><span style="color: #7f7f7f; text-ded
│   │    </span>pbar.update(<span style="color: #0000ff; text-decoration-cc
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
```

```
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #bfbf7f">/data/source/langchain/langchain/document_loaders/</span><
decoration-color: #808000; font-weight: bold">directory.py</span>:<span st
color: #0000ff">78</span> in <span style="color: #00ff00; text-decoration-
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f">75 </span><span style="color: #7f7f7f; text-de
</span><span style="color: #0000ff; text-decoration-color: #0000ff">if</sp
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f">76 </span><span style="color: #7f7f7f; text-de
</span><span style="color: #0000ff; text-decoration-color: #0000ff">if</sp
style="color: #ff00ff; text-decoration-color: #ff00ff">or</span> <span sty
color: #00ffff">self</span>.load_hidden:                  <span style
#800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f">77 </span><span style="color: #7f7f7f; text-de
</span><span style="color: #0000ff; text-decoration-color: #0000ff">try</s
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #800000">❱ </span>78 <span style="color: #7f7f7f; text-decoration-
</span>sub_docs = <span style="color: #00ffff; text-decoration-color: #00f
style="color: #00ffff; text-decoration-color: #00ffff">str</span>(i), **<s
decoration-color: #00ffff">self</span>.loader_kwargs).load()     <span sty
color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f">79 </span><span style="color: #7f7f7f; text-de
│   </span>docs.extend(sub_docs)
decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f">80 </span><span style="color: #7f7f7f; text-de
</span><span style="color: #0000ff; text-decoration-color: #0000ff">except
text-decoration-color: #00ffff">Exception</span> <span style="color: #0000
#0000ff">as</span> e:                            <sp
decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>   <s
decoration-color: #7f7f7f">81 </span><span style="color: #7f7f7f; text-de
│    </span><span style="color: #0000ff; text-decoration-color: #0000ff">i
text-decoration-color: #00ffff">self</span>.silent_errors:
style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #bfbf7f">/data/source/langchain/langchain/document_loaders/</span><
```

```
decoration-color: #808000; font-weight: bold">text.py</span>:<span style='
#0000ff">44</span> in <span style="color: #00ff00; text-decoration-color:
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>        <s
decoration-color: #7f7f7f">41 </span><span style="color: #7f7f7f; text-ded
│   </span><span style="color: #0000ff; text-decoration-color: #0000ff">ex
text-decoration-color: #00ffff">UnicodeDecodeError</span>:
style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">42 </span><span style="color: #7f7f7f; text-ded
│   │    </span><span style="color: #0000ff; text-decoration-color: #0000ff
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">43 </span><span style="color: #7f7f7f; text-ded
</span><span style="color: #0000ff; text-decoration-color: #0000ff">else</
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span> <spa
color: #800000">❱ </span>44 <span style="color: #7f7f7f; text-decoration-c
</span><span style="color: #0000ff; text-decoration-color: #0000ff">raise<
decoration-color: #00ffff">RuntimeError</span>(<span style="color: #80800(
loading {</span><span style="color: #00ffff; text-decoration-color: #00ff1
style="color: #808000; text-decoration-color: #808000">}"</span>) <span st
color: #0000ff">from</span> <span style="color: #00ffff; text-decoration-c
underline">e</span>                     <span style="color: #800000; text-decoratio
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">45 </span><span style="color: #7f7f7f; text-ded
</span><span style="color: #0000ff; text-decoration-color: #0000ff">except
text-decoration-color: #00ffff">Exception</span> <span style="color: #0000
#0000ff">as</span> e:
decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">46 </span><span style="color: #7f7f7f; text-ded
</span><span style="color: #0000ff; text-decoration-color: #0000ff">raise<
decoration-color: #00ffff">RuntimeError</span>(<span style="color: #80800(
loading {</span><span style="color: #00ffff; text-decoration-color: #00ff1
style="color: #808000; text-decoration-color: #808000">}"</span>) <span st
color: #0000ff">from</span> <span style="color: #00ffff; text-decoration-c
underline">e</span>                      <span style="color: #800000; text-decor
<span style="color: #800000; text-decoration-color: #800000">│</span>    <s
decoration-color: #7f7f7f">47 </span>
<span style="color: #800000; text-decoration-color: #800000">│</span>
<span style="color: #800000; text-decoration-color:
#800000">└─────────────────────────────────────────────────────────
<span style="color: #ff0000; text-decoration-color: #ff0000; font-weight:
..<span style="color: #800080; text-decoration-color:
```

```
 #800080">/../../../../tests/integration_tests/examples/</span><span style=
 #ff00ff">example-non-utf8.txt</span>
 </pre>
```

The file `example-non-utf8.txt` uses a different encoding the `load()` function fails with a helpful message indicating which file failed decoding.

With the default behavior of `TextLoader` any failure to load any of the documents will fail the whole loading process and no documents are loaded.

B. Silent fail

We can pass the parameter `silent_errors` to the `DirectoryLoader` to skip the files which could not be loaded and continue the load process.

```
loader = DirectoryLoader(path, glob="**/*.txt", loader_cls=TextLoader,
silent_errors=True)
docs = loader.load()
```

```
    Error loading
../../../../../tests/integration_tests/examples/example-non-utf8.txt
```

```
doc_sources = [doc.metadata['source']  for doc in docs]
doc_sources
```

```
['../../../../../tests/integration_tests/examples/whatsapp_chat.txt',
     '../../../../../tests/integration_tests/examples/example-
utf8.txt']
```

C. Auto detect encodings

We can also ask `TextLoader` to auto detect the file encoding before failing, by passing the `autodetect_encoding` to the loader class.

```
text_loader_kwargs={'autodetect_encoding': True}
loader = DirectoryLoader(path, glob="**/*.txt", loader_cls=TextLoader,
loader_kwargs=text_loader_kwargs)
docs = loader.load()
```

```python
doc_sources = [doc.metadata['source']  for doc in docs]
doc_sources
```

```
    ['../../../../../tests/integration_tests/examples/example-non-
utf8.txt',

'../../../../../tests/integration_tests/examples/whatsapp_chat.txt',
     '../../../../../tests/integration_tests/examples/example-
utf8.txt']
```