File parser for Tolstoy’s War & Peace

The unstructured file is parsed using Python 2.7 and the result is stored as a nested dictionary which can be converted to JSON as needed.

Read file, Book Name

Read the large text file by line enumeration instead of reading in chunks into the memory.

With Open(filename, ‘r’):

For each line you enumerate, check if there’s a‘Book Name’. If yes, then initialize the outer most nested dictionary. If no ‘Book Name’ is found, assign it with None

Result = {‘Book Name’:{}}

Default: result = {None:{}}

Read Chapters

For each line, if there is a line that starts with ‘CHAPTER’, initialize the 2nd level of the nested dictionary. Set In_Chapter flag to TRUE, Para_lines = []

Result = {‘Book Name:{‘Chapter Name’:{}}

Default = {‘Book Name:{{}}

In Chapter, process Paragraph

When the line being read is inside a chapter, if there's no empty line, then start accumulating the lines in a paragraph into the list para_lines.

```
para_lines.append(line)
```

If there's a empty line encountered, and para_lines array is non-empty, then it means that it line reached the end of a paragraph. Hence, count para_lines, process the paragraph.

```
result[book_num_yr][chapter_index][para_id] =
self.parseParagraph(para_lines)
```

Process words

While processing paragraph, pass each sentence through parseWords() function to get the alphanumeric words, along with word id.

result[sid] = {sentence: self.parseWords(sentence)}

result[wid] = re.sub('[^a-zA-Z0-9 \n\.]', '', word)

Use regex to get the alphanumeric characters only.

Final nested struct: result= {‘Book Name’:{‘Chapter Name’:{‘para_id:{‘sentence_ID:val {word_id:val}}}}