

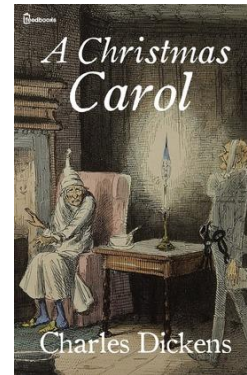
Advanced Natural Processing

ASSIGNMENT 3

Team: Prashanth Sekar(PRTHIRU), Pravin Sundar(PSUNDAR), Barathwaaj Parthasarathy(BPARTHA), Archish Ramesh Babu(ARAMESHB)

Text: “The Monkey’s Paw” (1902) by W.W. Jacobs & “Christmas Carol” (1843) by Charles Dickens

1. Report which texts you're using and to what extent they are similar to or different from each other. Make sure the text contains dialogue in some fashion.



We chose “The Monkey’s Paw” (1902) by W.W. Jacobs as our training dataset and we chose “Christmas Carol” (1843) by Charles Dickens as our testing data set. Although both novels are 60 years apart they have a similar narration and dialogue structure. In both the cases, conversations between characters are clearly distinguished with line breaks and the dialogues are enclosed within quotes.

The story of Monkey’s Paw comprises of a very limited number of characters, a macabre story that revolves around a family and few others. On the other hand, Christmas Carol is a very elaborate story of a miserly man whose greedy and cold-hearted approach to life is reversed, involving lot of characters with a variety of personalities.

Both novels start off with a clear introduction to the scene and characters followed by conversations between characters. Both classics are written by English authors and classics, and they are similar in language in the sense both Dickens and Jacobs characters use colloquial language.

On looking at the text we identified key patters like “said Mrs.White”, “he said”, “repeated Mr. White”, because they indicate who is speaking. As these indicators give the readers a good understanding during character conversations. Ideally, we will also use similar approach to identify and extract speak(X,Y) relations.

2. Comb over the development text, with the Stanford (and optionally the other pipelines) annotations added, in order to identify (manually, semi-automatically, and eventually fully automatically) when one character is speaking to another. Specifically, your goal is to extract relations of the form $Speak(X,Y)$, where X and Y are characters. (You are allowed to leave X and/or Y unspecified in some cases, noting that these are cases you would eventually like to make fully specific.)

Manually:

The way we identify who is talking to whom is straight forward. While reading a book, it is clearly stated either before the dialogue or after the dialogue. This can be observed clearly by considering this small text:

"I suppose all old soldiers are the same," said Mrs White. "The idea of our listening to such nonsense! How could wishes be granted in these days? And if they could, how could two hundred pounds hurt you, father?"

"Might drop on his head from the sky," said the frivolous Herbert.

In the above paragraph from part two, we can say the first dialogue is said by Mrs. White as it is directly mentioned after the phrase. From the second line we can see that Mrs. White is talking to Herbert. With such instances we will be able to identify $Speak(X,Y)$. In this book we know there are only a countable number of characters and hence we will be able to formulate the total relations as well.

This is a time-consuming but a simple way to check the whole book for such relations. For long books, this approach will be very tedious procedure and will most definitely result in human error due to negligence and stress. Also, in cases where the characters are mentioned early in the text, this strategy of jumping directly to dialogues will fail.

Semi – Automatically:

For this approach, we first analyze the sentence the structure for the book

A few examples from the Book:

"Likely," said Herbert, with pretended horror. "Why, we're going to be rich, and famous and happy. Wish to be an emperor, father, to begin with; then you can't be henpecked."

"Well, I don't see the money," said his son as he picked it up and placed it on the table, "and I bet I never shall."

"It must have been your fancy, father," said his wife, regarding him anxiously.

Based on the sentence structure we proceeded with creating an algorithm to identify the locations of text where there might be conversations by looking at double quotes primarily, we have also used other identifiers like *single quotes*, *said*, *told*, *explained*, *shouted* to improve the robustness of the algorithm.

This will create a list of text chunks, instead of the whole text where we had to search for conversations in the manual approach. Then we identify possible characters based on keywords(said, replied etc.) before and after the conversations identified by the previous step

Finally, we manually identify X and Y from these chunks and label X and Y. Also, we delete the characters who are not present in that chunk from the list generated by our algorithm

This approach is an effective approach because it uses the code to do the tedious work of combing over the entire book and uses the human intuition which the system lacks. This approach might possibly give better results than the fully automatic approach since the human intervention will solve a lot of ambiguities where an completely automated code might not. For example, in a case where some speaker speaks to an audience, the automatic approach will stumble. Having said that, this is still not the quickest way to tackle this problem and the code will invariably miss some isolated conversations.

Automatically:

In the automatic approach we utilized the power of Stanford Core NLP Tool. On running the book as a single file through the tool we obtained an xml file as the output(Fig 1: Snippet of the code run on the Stanford NLP Tool). The tool looks at each word in the book and identifies its lemma, Character position, Part of Speech, Speaker, NER and Sentiment.

```
F:\Indiana University\Fall 2018\ANLP\Stanford Core NLP\stanford-corenlp-full-2018-02-27>java --add-modules java.se.ee -cp "*" -Xmx10g edu.stanford.nlp.pipeline.StanfordCoreNLP -annotators "tokenize,ssplit,pos,lemma,ner,parse,dcoref" -file monkey-paw.txt -outputFormat xml
```

Sentence #27

Tokens

Id	Word	Lemma	Char begin	Char end	POS	NER	Normalized NER	Speaker	Sentiment
1	"	"	3268	3269	"	O			
2	He	he	3269	3271	PRP	O		PER15	
3	do	do	3272	3274	VBP	O		PER15	
4	n't	not	3274	3277	RB	O		PER15	
5	look	look	3278	3282	VB	O		PER15	
6	to	to	3283	3285	TO	O		PER15	
7	have	have	3286	3290	VB	O		PER15	
8	taken	take	3291	3296	VCN	O		PER15	
9	much	much	3297	3301	JJ	O		PER15	
10	harm	harm	3302	3306	NN	O		PER15	
11	,	,	3306	3307	,	O		PER15	
12	"	"	3307	3308	"	O		PER0	
13	said	say	3309	3313	VBD	O		PER0	
14	Mrs.	Mrs.	3314	3318	NNP	O		PER0	
15	White	White	3319	3324	NNP	PERSON		PER0	
16	,	,	3324	3325	,	O		PER0	
17	politely	politely	3326	3334	RB	O		PER0	
18	.	.	3334	3335	.	O		PER0	

Parse tree

(ROOT (S ("") (S (NP (PRP He)) (VP (VBP do) (RB n't) (VP (VB look) (S (VP (TO to) (VP (VB have) (VP (VCN taken) (NP (JJ much) (NN harm)))))))))) (,) (") (VP (VBD said)) (NP (NP (NNP Mrs.) (NNP White)) (,) (ADVP (RB politely))) (.)))

Sentence #140

Tokens

Id	Word	Lemma	Char begin	Char end	POS	NER	Normalized NER	Speaker	Sentiment
1	`	`	12059	12060	`	O			
2	I	I	12060	12061	PRP	O		753	
3	dare	dare	12062	12066	VBP	O		753	
4	say	say	12067	12070	VB	O		753	
5	,	,	12070	12071	,	O		753	
6	"	"	12071	12072	"	O		PER0	
7	said	say	12073	12077	VBD	O		PER0	
8	Mr.	Mr	12078	12081	NNP	O		PER0	
9	White	White	12082	12087	NNP	PERSON		PER0	
10	,	,	12087	12088	,	O		PER0	

- We can identify the speaker from the '*NER and Speaker*' results obtained from the NLP tool
- We then identify sentences based on Full Stops
- For these identified sentences we can use NER method to identify who spoke to whom
- In cases where we are able to pick up only one person from the sentence we look at previous occurrences from NER to identify the other person in the conversation and if we don't pick up any NER from previous text then we look for a NER from text post the respective conversation

By using this approach we identify the speaker and participant for each quoted sentence and if there are multiple other characters who are a part of the conversation with a single speaker then we create separate instances for each of the participating characters with the speaker.

3. Run your CoreNLP processing pipeline over the test data. Report in a table the total frequencies with which, according to your system, the different characters are speaking to each other

We decided to proceed with the Manual System due to the difficulties in resolving the ambiguities present in the semi-automatic and fully automatic approaches. We picked Chapter I from “The Monkey’s Paw” as a train data set and STAVE III from “A Christmas Carol” as our test data set.

The Monkey's Paw		
Speaker	Listener	Frequency
Morris	Mr. White	9
Morris	Mrs White	6
Mr. White	Morris	6
Mr. White	Hebert	5
Hebert	Mr. White	4
Mrs White	Morris	3
Hebert	Morris	2
Mr. White	Mrs White	2
Mrs White	Hebert	2
Mrs White	Mr. White	2
Hebert	Mrs White	1

A Christmas Carol		
Speaker	Listener	Frequency
Ghost Present	Scrooge	14
Scrooge	Ghost Present	14
Scrooge's Nephew	Scrooge's Neice	8
Scrooge's Neice	Scrooge's Nephew	8
Bob	Mrs Cratchit	7
Mrs Cratchit	Bob	5
Mrs Cratchit	Martha	3
Scrooge's Nephew	Scrooge	2
Scrooge	Scrooge's Nephew	2
Mrs Cratchit	Peter	1
Belinda	Mrs Cratchit	1
Martha	Mrs Cratchit	1
Peter	Martha	1
Bob	Tim	1
Tim	Bob	1

4. As a final part of the report, add a note on what you would do if you had more time to make the system better

To improve our system to identify and extract the speak(X,Y) relations, we will need to consider all the anomaly cases for our algorithm that are observed in the books.

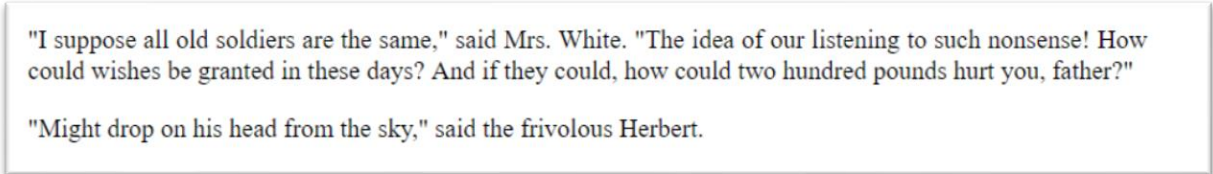
Places for Improvement:

- a. There are multiple cases where a couple of people are speaking but many other characters could be part of the conversation without speaking. Our algorithm currently fails to account for this
 - a. We would need to look at previous sentences and narrator comments to better grouped tag the characters who are a part of the conversation
- b. For sentences like *"He went away" said the old man*. Our approach will only tag the speaker as old man but will fail to mention the old man's name or other details. In the below picture the old man would refer to Mr.White



"I wish for two hundred pounds," said the old man distinctly.

- a. We would need to scan through the entire text to better understand the character referred as old man
- c. Identifying end of chapters will help in better tagging the characters with respect to transition of scenes
 - a. We can look at Roman Numerals or the combination of the word *Chapter* followed by roman letter to identify end of chapter
- d. If a character speaks a lot of dialogs and if the name of the speaker is mentioned at the middle then our algorithm will incorrectly tag the speaker for the second half of that dialog. In the below image the speaker of the sentence *"The idea of our listening..."* is spoken by Mrs. White but our algorithm will tag the speaker as Herbert, because his will be the first name to appear after that particular text
 - a. We will need to identify speaker also based on words spoken



"I suppose all old soldiers are the same," said Mrs. White. "The idea of our listening to such nonsense! How could wishes be granted in these days? And if they could, how could two hundred pounds hurt you, father?"

"Might drop on his head from the sky," said the frivolous Herbert.

- e. Further running the system across many books to analyze failed speak(X,Y) relations to further improve the algorithm and technique.