CharBoxes: A System for Automatic Discovery of Character Infoboxes from Books*

Manish Gupta
Microsoft, India
gmanish@microsoft.com

Piyush Bansal
IIIT, Hyderabad, India
piyush.bansal@research.iiit.ac.in

Vasudeva Varma
IIIT, Hyderabad, India
vv@iiit.ac.in

ABSTRACT

Entities are centric to a large number of real world applications. Wikipedia shows entity infoboxes for a large number of entities. However, not much structured information is available about character entities in books. Automatic discovery of characters from books can help in effective summarization. Such a structured summary which not just introduces characters in the book but also provides a high level relationship between them can be of critical importance for buyers. This task involves the following challenging novel problems: 1. automatic discovery of important characters given a book; 2. automatic social graph construction relating the discovered characters; 3. automatic summarization of text most related to each of the characters; and 4. automatic infobox extraction from such summarized text for each character. As part of this demo, we design mechanisms to address these challenges and experiment with publicly available books.

Categories and Subject Descriptors

H.2.8 [**Information Systems**]: Database Applications—*Data Mining*; H.4.0 [**Information Systems Applications**]: General

General Terms

Algorithms, Design, Experimentation

Keywords

Analysis of Books, Character Infoboxes, Person disambiguation, Person-person network

1. INTRODUCTION

Given a book, can we automatically discover all person mentions in the book, and organize them in an interesting way? Such a person knowledge base could be important in helping users understand the characters in the book better. It could also serve as a repository that connects books with different characters (or persons). The characters could be fictional or real. In this demo, we will showcase a system `CharBoxes` which not just discovers all characters from books but also extracts details about each character using deep natural language processing and some domain-specific heuristics.

*The first two authors have equal contribution.

SIGIR'14, July 6–11, 2014, Gold Coast.
ACM 978-1-4503-2745-9/14/04.

To the best of our knowledge, the proposed system is the first to focus on extraction of character infoboxes from books. The system takes the book text as an input and outputs an infobox for every character which includes the following about the character. We provide examples for "Harry Potter" in the book "Harry Potter and the Sorcerer's Stone."

- Most related persons (along with the relationship), e.g., Hermione Granger (friend), Ron Weasley (friend)
- Most related places and organizations (along with verbs indicating relation), e.g., Hogwarts, Gringotts
- Year of birth/death or age, e.g., 11 years
- Personality traits of the person, e.g., "loyal and kind hearted, wears glasses, is thin, and has his mother's green eyes, scar located on his forehead"
- Overall sentiment of the person (hero/villian), e.g., "hero"
- Frequently mentioned facts, e.g., "plays Quidditch, is a wizard, uses the Cloak"
- Sociability of the person, e.g., 0.8
- Books in which appeared, e.g., "Harry Potter and the Sorcerer's Stone", "Harry Potter and Chamber of Secrets", ...
- Character-centric text summary, e.g., "To Harry Potter – the boy who lived! Harry had a thin face, knobbly knees, black hair, and bright green eyes. He wore round glasses held together with a lot of Scotch tape because of all the times Dudley had punched him on the nose. The only thing Harry liked about his own appearance was a very thin scar on his forehead that was shaped like a bolt of lightning ..."

2. RELATED WORK

Our work is related to the following two areas of research: "Analysis of books or multi-documents" and "Extracting structured data from free text". There is very little work in the area of analysis of books. Most of the previous work has focused on summarization of books [4, 6]. There has been quite some work on extracting structured data from free text [1, 2, 7, 8]. The proposed system is the first to focus on extraction of character infoboxes from books. Once we extract character-specific text from the book, we use a novel sentiment analysis based summarizer besides the summarization mechanism discussed in [5]. The character-specific summary is then used to extract subject-predicate-object facts. Also, a few patterns as discussed in [2] are designed to extract various structured attributes about the characters from the character-specific summary.

3. SYSTEM COMPONENTS

Figure 1 shows a system diagram for `CharBoxes`. The system has two main components: Linguistic Analysis and Character-Infobox Generation.

3.1 Linguistic Analysis

Characters Extraction

The system takes the text of a book as input. First the system applies basic heuristics to extract the author and the year of publi-

cation of the book. The book text is then parsed using a POS Tagger and a NER to obtain person, place and organization entities. Person names are sorted by frequency of occurrence in the book text. Person entities are further cleaned using simple heuristics like removing names with frequency less than a threshold, merging names considering different variations, etc. This gives us a list of all characters in the book.

Co-reference Resolution and Dependency Analysis

For deeper analysis related to these characters, first the chapter boundaries need to be detected. For books with clear chapter boundaries, hints are obtained from word clues like "Chapter X" or "Lesson X," and also from the table of contents. If page boundaries are available, an incompletely filled page also denotes a chapter boundary. For books without any clear page boundaries or chapter boundaries, TextTiling algorithm [3] is used for topic shift detection. Co-reference resolution is performed on each chapter followed by Stanford dependency analysis[1]. Co-reference resolution ensures that all mentions of the person names get resolved. This processing results into a linguistically analyzed book text.

3.2 Character-Infobox Generation

Person-Person Graph Construction

This text is used to build a person-person graph based on dialogue extraction (using presence of words like "say, said, tell, told, screamed"). Next, person name disambiguation is performed using the graph. The text may refer to some character using the last name, e.g., "Weasley" in "Harry Potter and the Philosopher's Stone". Based on the context, a human can disambiguate this "Weasley" to "Ron Weasley". Such a disambiguation is essential for accurate understanding of sentences. Hence, disambiguation is performed for last names based on: (1) Context words, (2) Mention of full name in vicinity, (3) Frequency of co-occurrence with other entities in the vicinity based on the graph. Once disambiguation is done, the resolution is used to refine the person-person graph also. The person-person graph is further annotated with a few select relationships that can be inferred using relationship words observed in the context of the two persons (mother, father, sibling), or using dialogues between the two persons (friend/enemy).

Related Places and Organizations Extraction

Given a character, all mentions of the character are identified. Next, most frequent places and organizations associated with the character are discovered by scores them based on frequency and distance from the character mentions. Also, the linking verb is used to establish the relationship of the character with the place or the organization. For example, "studies" could be the most frequent verb linking "Harry Potter" with "Hogwarts." Thus, we obtain most frequent related places and organizations for every character and also define the type of relationship using the connecting verb.

Character-Centric Summary Generation

All sentences which mention a character are collected to get all text related to a character. This text is processed to extract subject, predicate and object triplets. These triplets are then summarized using a mix of two mechanisms: (1) using the semantic subgraph using triplets as discussed in [5], and (2) based on the strength of the sentiments around that sentence. The intuition is that sentences with extremely sentimental context are important.

Character-Centric Facts Extraction

The character-specific summary is further analyzed to obtain the following information about the character: (1) year of birth/death, (2) looks, qualities of the person (either direct text mentions or inferred from the spoken sentences), (3) overall sentiment of the person (hero/villian), (4) frequently mentioned facts (like relation between "Harry Potter" and "quidditch" linked by the verb "plays"), (5) sociability of the person (based on number of other characters it interacts with).

[1]http://nlp.stanford.edu/software/stanford-dependencies.shtml

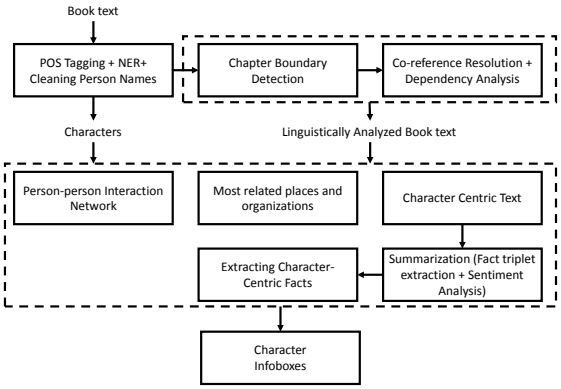

Figure 1: `CharBoxes`: System for Infobox Extraction for Characters from Books

4. DEMONSTRATION

We will demonstrate the capabilities of `CharBoxes` as described in Section 3. We have collected books from the Project Gutenberg[2]. The demo will allow the users to select any book and see online analysis of the book. The system shows the extracted details for the book (year of publication and authors, if available) and the list of characters with their frequency of occurrence on the first page. The user can click on any character to see the Infobox extracted for it from the book. Since the processing of long text takes time, we can showcase real time results for short books. But we will have already processed results ready for books in our dataset.

5. CONCLUSION

We present the system `CharBoxes` which takes a book as input and outputs structured Infoboxes for various characters in the book. The system utilizes deep natural language processing techniques complemented by domain specific heuristics. The system can be very useful in summarizing books in a structured way in terms of insights about characters discussed in the book.

6. REFERENCES

[1] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R Hruschka Jr, and Tom M Mitchell. Toward an Architecture for Never-Ending Language Learning. In *AAAI*, 2010.

[2] Oren Etzioni, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. Web-scale Information Extraction in KnowItAll: (Preliminary Results). In *WWW*, pages 100–110, 2004.

[3] Marti A. Hearst. Multi-paragraph Segmentation of Expository Text. In *ACL*, pages 9–16, 1994.

[4] Anna Kazantseva and Stan Szpakowicz. Summarizing Short Stories. *Computational Linguistics*, 36(1):71–109, Mar 2010.

[5] Jure Leskovec, Natasa Milic-Frayling, Marko Grobelnik, and J Leskovec. Extracting Summary Sentences based on the Document Semantic Graph. *Microsoft Research, Microsoft Corporation*, 2005.

[6] Rada Mihalcea and Hakan Ceylan. Explorations in Automatic Book Summarization. In *EMNLP*, pages 380–389, 2007.

[7] Stephen Soderland. Learning Information Extraction Rules for Semi-Structured and Free Text. *Machine Learning*, 34(1-3):233–272, 1999.

[8] Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: A Large Ontology from Wikipedia and Wordnet. *Web Semantics*, 6(3):203–217, 2008.

[2]http://www.gutenberg.org/