NovelPersepective

Identifying point of view characters

Lyndon White, Roberto Togneri, Wei Liu, Mohammed Bennamoun

THE UNIVERSITY OF WESTERN AUSTRALIA

What? Why? Why are you doing this to books?

Many novels, especially epic fantasy series, are written from the Point of View (POV) of many different characters.

They feature parallel sub-stories tracking the journey of each POV character.

Readers sometimes wish to read just one character's story; for example, on a second read through.

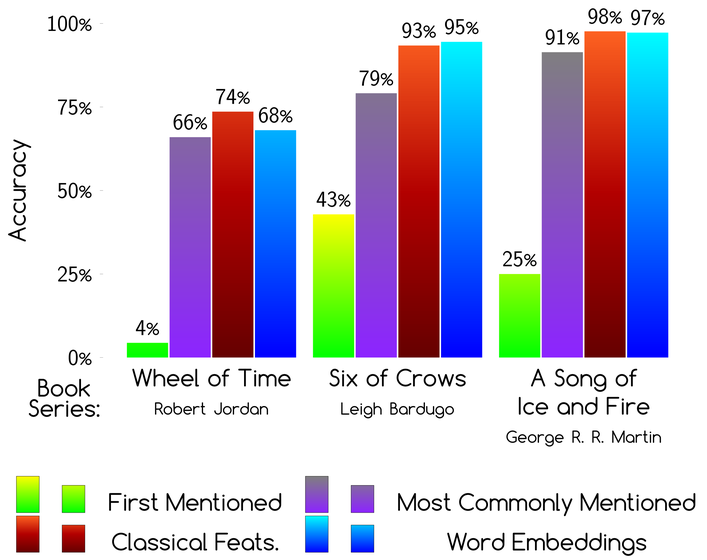
We have made a tool that allows the user to slice-up and restitch their ebooks around each POV character.

The challenging part is that most books do not label the sections with the name of the POV character, rather the reader works it out.

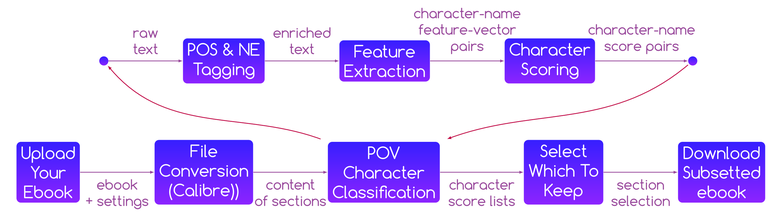
The source code is publicly available

MIT Licensed
https://github.com/oxinabox/NovelPerspective
Built on CherryPy, NLTK, Scikit-Learn, EbookLib and Calibre

Results on popular series of novels



The process for subsetting ebooks by POV



Baseline methods for determining POV

First Mentioned Named Entity

Features: first occurrence of named entity token in the section.
Scoring: earliest mentioned scores highest, $S_i = 2^{-rank(f_i)}$
Result: terrible. Other named entities often occur before POV.

Most Commonly Mentioned Named Entity

Features: count of occurrences of named entity token in section.
Scoring: most mentioned scores highest, $S_i = \frac{f_i}{\sum_{\forall j} f_j}$
Result: generally solid, but fooled by descriptions focusing on others.

Machine learning methods for determining POV

Classical Features + Logistic Regression

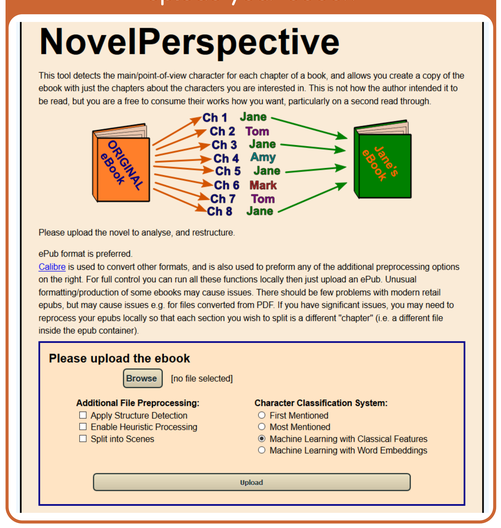
Features: position, and occurrence frequency, plus parts of speech co-occurring frequency. Total 200 dims.
Scoring: use logistic regression model on if POV or not, $S_i = \frac{P(f_i)}{\sum_{\forall j} P(f_j)}$
Result: generally great. Main characters occur near verbs and grammar. This gives an edge over frequency information alone.

Word Embeddings + RBF-SVM

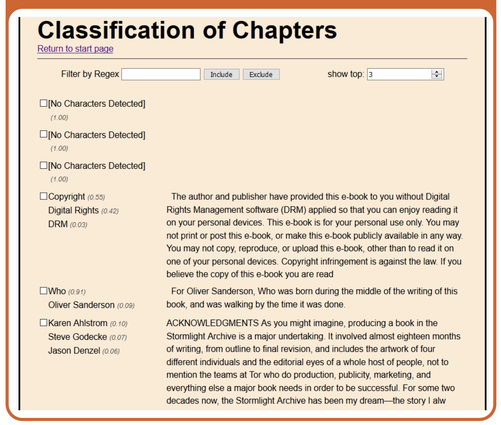
Features: concatenation of FastText word embeddings for the adjacent words, averaged over all occurrences. Total 600 dims.
Scoring: use RBF-SVM model on if POV or not, $S_i = \frac{P(f_i)}{\sum_{\forall j} P(f_j)}$
Result: generally great. However due to high dimensionality, this method needs a lot of training data from other labelled books.

What does it look like?

Upload your ebook

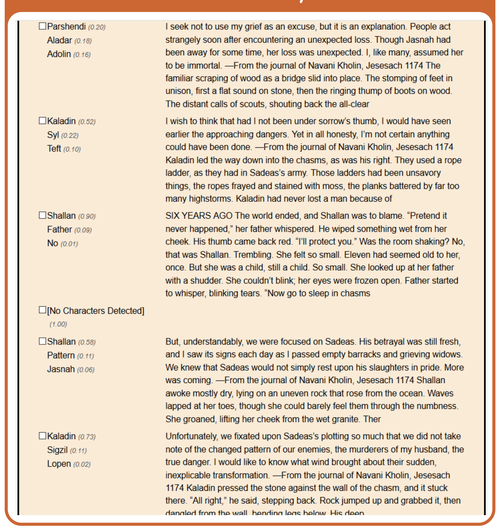


Select which sections to keep



Demonstration on *"Words of Radiance"* by Brandon Sanderson.
Using the classical features method.

Sections are labelled by POV & score



http://novelperspective.ucc.asn.au/