An Extraction and Representation Pipeline for Literary Characters

Funing Yang

Wellesley College
Wellesley, Massachusetts 02481
fyang3@wellesley.edu

Abstract

Readers of novels need to identify and learn about the characters as they develop an understanding of the plot. The paper presents an end-to-end automated pipeline for literary character identification and ongoing work for extracting and comparing character representations for full-length English novels. The character identification pipeline involves a named entity recognition (NER) module with F1 score of 0.85, a coreference resolution module with F1 score of 0.76, and a disambiguation module using both heuristic and algorithmic approaches. Ongoing work compares event extraction as well as speech extraction pipelines for literary characters representations with case studies. The paper is the first to my knowledge that combines a modular pipeline for automated character identification and representation extraction and comparisons for full-length English novels.

Introduction

Literary characters play an integral role in plot development and narrative understanding. The digitization of literary databases and the advent of automated information retrieval have enabled computational character extraction and understanding, from a hierarchical Bayesian model for character types inference in 18th and 19th-century English novels (Bamman, Underwood, and Smith 2014), to a 'narratologically' grounded definition of character and a supervised classifier to identify characters in Russian folktales (Jahan and Finlayson 2019). However, there has been no work on an end-to-end character identification pipeline for full-length novels with the most recent natural language processing modeling techniques, and few have focused on extracting and comparing dense vector representations of literary characters from narratology grounded definitions. This paper has the following two contributions: 1) A modular pipeline to automatically extract literary characters from an unstructured text of a full-length novel that achieves reasonable performance for Jane Austen's *Sense and Sensibility*. 2) Quantified embedding representation extractions and comparisons for literary characters from narrative theory's definitions of literary events and speech.

Named Entity Recognition

The first module in the character identification pipeline involves extracting entities from an unstructured text with a Named Entity Recognition (NER) model. As the literary domain poses unique challenges given varying styles and structures, the LitBank NER dataset that covers 100 English novels (Bamman, Popat, and Shen 2019) was chosen for model training. While LitBank collects up to 4 nested layers of entity labels for each token, to obtain the most information for each mention, only the outermost nested mention layer (with the longest spans) is collected.

I have trained and compared 4 separate model architectures: 1) CRF (baseline) with word identity (e.g. case, digit), word suffix, word shape, and part of speech tags; 2) LSTM; 3) Bi-LSTM-GloVe consisting of an embedding layer, which is then passed through a dropout layer and a Bi-LSTM layer; 4) Bi-LSTM-CRF-GloVe that combines previous attempts with CRF-Bi-LSTM (Huang, Xu, and Yu 2015) with its proven robustness and inexpensiveness to train; I also applied Glove (Pennington, Socher, and Manning 2014) word embeddings for introducing context and replaced rare class labels (i.e., VEH (vehicle) and ORG (organization)) unimportant for performance to 'O' tags. Overall, the weighted average of precision, recall, and F1 score for each class are used to evaluate the NER model as an aggregated overall performance below on the Litbank dataset:

	Recall	Precision	F1
CRF	0.51	0.67	0.58
LSTM	0.83	0.79	0.79
BiLSTM	0.82	0.82	0.81
BiLSTM-CRF	0.86	0.87	0.85

Table 1: Averaged results for NER models

The Bi-LSTM-CRF model with GloVe embedding and rare label classes removed achieved F1 score of 0.85 and outperforms the CRF baseline by over 20 percentage points, and was used for the downstream pipeline.

Coreference Resolution

Coreference Resolution is the second module in the character identification pipeline where it links expressions that refer to the same entity. Toward the goal of improving perfor-

	Recall	Precision	F1
SpanBERT + Linear	44.1	56.9	48.3
SpanBERT + Bi-LSTM	72.4	72.9	72.1
BART + Bi-LSTM	61.9	55.9	40.7
SpanBERT + Bamman et al.	76.8	75.7	76.1
SpanBERT + GRU + Bamman et al.	77.7	76.0	76.1

Table 2: CONLL score for Coreference Models

mance on the LitBank coreference resolution dataset (Bamman, Lewke, and Mansoor 2020), I built my pipeline on top of Bamman et. al's pre-processing and post-processing pipeline but with 5 different model architectures for a comparative study on model performance and compared different contextual embeddings (SpanBERT and BART), and classification heads (linear, Bi-LSTM, GRU). First, I tried SpanBERT embeddings with linear layer (baseline), as SpanBERT outperforms BERT on coreference resolution tasks by masking spans of tokens instead of individual ones (Joshi et al. 2020). The mention representation consisted solely of the raw SpanBERT embeddings for the start and end indices, and the mention representations get passed through a linear layer with dropout to compute the candidate scores. Second, I tried SpanBERT Embeddings with Bi-LSTM, as Bi-LSTMs can capture the state history over a wide context both to the left and right of a given word. Third, I tried BART (Lewis et al. 2020) embeddings with Bi-LSTM architecture. Fourth, I tried SpanBERT embeddings with the Bamman et al. pipeline. And fifth, I tried SpanBERT embeddings + GRU with the Bamman et al. pipeline. The presence of 4) and 5) are to explore whether complex literature-specific embeddings and attention in the original pipeline are useful for actual predictions. Surprisingly, my simpler setup is almost at par with Bamman's more complex architectures.

For model evaluation, I computed precision, recall, and F1 score for all the coreference architectures above with the CONLL score composed of three well-established metrics: B-CUBED, MUC, CEAF (Moosavi and Strube 2016) in Table 2. The SpanBERT + GRU architecture achieved the top performance of 77.7 recall, 76.0 precision, and 76.1 F1, which is applied downstream.

Evaluation

While a quantitative evaluation of the full pipeline is still in-progress, I have qualitatively evaluated the pipeline on Jane Austen's *Sense and Sensibility* as well as J.K. Rowling's *Harry Potter* to test generalization for two contrasting genres. Consider the following predictions where an italicized span represents a different character cluster:

> *The family of Dashwood had long been settled in Sussex*. The late owner of this estate was a single man, who lived to a very advanced age, and who for many years of his life, had *a constant companion* and *housekeeper in his sister*. (Austen *Sense and Sensibility*)

In summary, the model produces largely sensible results consistent with maximal span tagging in NER and coreference. The most common failure cases are that NER often fails to recognize uncommon names as entities (such as 'Elinor'); that ample use of descriptors and nested mentions make resolving entities difficult; and very prominent entities representing main characters tend to appear in multiple clusters because the coreference model has trouble recognizing links that are distant from each other. However, I observed improvements with downstream finetuning with heuristics and word embedding comparisons.

Ongoing Work: Character Representation

Aside from the quantitative evaluation framework for the character identification pipeline, my ongoing work involves a comparison on character representation from two narratology definitions: 1) character as a representation of literary events; 2) character being represented by their direct speech. I have built an unsupervised event extraction pipeline inspired from (Chambers and Jurafsky 2008) that mines event clusters per literary character; a representation extraction and comparison pipeline for both contextualized embeddings (i.e., BERT) and static embeddings (i.e., GloVe) and uses clustering and distance metrics for characters comparison. I am also designing a speech-based representation pipeline to contrast characters from their actions versus speech.

Acknowledgements

Thanks to my advisor Professor Carolyn Anderson for her mentorship; to Professor Eni Mustafaraj and Professor Peter Mawhorter for feedback; to Chole Lee, Olivia Pfeiffer, and Sanjana Narayanan for help with character identification.

References

Bamman, D.; Lewke, O.; and Mansoor, A. 2020. An annotated dataset of coreference in English literature. In *Proc. LREC*.

Bamman, D.; Popat, S.; and Shen, S. 2019. An annotated dataset of literary entities. In *Proc. NAACL*.

Bamman, D.; Underwood, T.; and Smith, N. A. 2014. A Bayesian mixed effects model of literary character. In *Proc. ACL*.

Chambers, N.; and Jurafsky, D. 2008. Unsupervised learning of narrative event chains. In *Proc. ACL*.

Huang, Z.; Xu, W.; and Yu, K. 2015. Bidirectional LSTM-CRF models for sequence tagging. arXiv:1508.01991.

Jahan, L.; and Finlayson, M. 2019. Character identification refined: A proposal. In *Proceedings of the First Workshop on Narrative Understanding*.

Joshi, M.; et al. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8: 64–77.

Lewis, M.; et al. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proc. ACL*.

Moosavi, N. S.; and Strube, M. 2016. Which coreference evaluation metric do you trust? A proposal for a link-based entity aware metric. In *Proc. ACL*.

Pennington, J.; Socher, R.; and Manning, C. 2014. GloVe: Global vectors for word representation. In *Proc. EMNLP*.