Annotating Characters in Literary Corpora: A Scheme, the CHARLES Tool, and an Annotated Novel

Hardik Vala[1], **Stefan Dimitrov**[1], **David Jurgens**[2], **Andrew Piper**[1], **Derek Ruths**[1]
[1]McGill University, [2]Stanford University
[1]Montreal, Canada, [2]Stanford, USA
{hardik.vala, stefan.dimitrov}@mail.mcgill.ca, jurgens@stanford.edu, {andrew.piper, derek.ruths}@mcgill.ca

Abstract

Characters form the focus of various studies of literary works, including social network analysis, archetype induction, and plot comparison. The recent rise in the computational modelling of literary works has produced a proportional rise in the demand for character-annotated literary corpora. However, automatically identifying characters is an open problem and there is low availability of literary texts with manually labelled characters. To address the latter problem, this work presents three contributions: (1) a comprehensive scheme for manually resolving mentions to characters in texts. (2) A novel collaborative annotation tool, CHARLES (*CHAracter Resolution Label-Entry System*) for character annotation and similiar cross-document tagging tasks. (3) The character annotations resulting from a pilot study on the novel Pride and Prejudice, demonstrating the scheme and tool facilitate the efficient production of high-quality annotations. We expect this work to motivate the further production of annotated literary corpora to help meet the demand of the community.

Keywords: Annotation tool, Collaborative annotation, Annotation scheme, Literary corpora

1. Introduction

> *"It begins with a character, usually, and once he stands up on his feet and begins to move, all I can do is trot along behind him with a paper and pencil trying to keep up long enough to put down what he says and does."* — William Faulkner

Character is fundamental to literary analysis, forming the basis of much computational research in literary domains. They are crucial to investigating various literary aspects: Social networks embedded in narratives, where characters form the nodes and their interactions and relationships form the edges, have been useful in determining character importance and role (Agarwal et al., 2012), understanding genre (Ardanuy and Sporleder, 2014), investigating the impact of setting on the social world described in novels (Jayannavar et al., 2015; Elson et al., 2010; Moretti, 2005), and even generating story text (Sack, 2012). Previous work has also attempted to induce character archetypes using the text associated to character mentions in large corpora (Bamman et al., 2014b; Bamman et al., 2014a). Elsner (2012) outlines representations of narrative centered around characters and their relationships, and then explores differences in various works through these abstractions.

Such works rely on the accurate identification of characters and their mentions, but state-of-the-art methods for automatically detecting and resolving character mentions perform poorly (Vala et al., 2015) and there is sparse availability of manually annotated datasets. Moreover, the growing application of computational methods in researching narrative works has been met with a growing demand for annotated literary corpora, especially for character-driven work (Agarwal et al., 2010; He et al., 2013). Previous works have used gold mentions (Agarwal et al., 2013; Lee and Yeung, 2012), or restrict to using only highly prevalent characters (Ardanuy and Sporleder, 2014), for which current automatic systems are more accurate, underscoring the need for efficient ways of building annotated literary corpora.

To address the scarcity of such corpora, we propose an annotation scheme and system for the task of character resolution, i.e. linking each mention of characters to the referent characters (a prerequisite to this task is detecting which mentions refer to characters). This paper offers the following contributions:

- A comprehensive annotation scheme for detecting and resolving character mentions.
- A novel, online and distributed annotation tool CHARLES[1] for carrying out such annotation tasks with multiple annotators simultaneously (not necessarily co-located).
- The complete list of character-resolved mentions of the novel, *Pride and Prejudice*, produced from a pilot annotation study which demonstrates the high quality of annotations.

Our system is designed for the character resolution task, which closely resembles the NLP task of cross-document entity coreference resolution, but we show how it can be easily extended to similiar annotation tasks.

The remainder of the paper is structured as follows: We outline the character resolution annotation scheme in Section 2., describe the annotation tool CHARLES in Section 3., discuss the results of the pilot study in Section 4., review related work in Section 5., and finish with concluding remarks and future work in Section 6.

2. Annotation Scheme

The goal of the character resolution annotation task is to identify the character, or set of characters, each *alias* references. An *alias* is simply the token or phrase mention of a possible character, or multiple characters, in text. They can take the form of proper names (e.g. *Mr. Bennet*), common

[1]https://charles.networkdynamics.org

nouns (e.g. *daughter*), and pronouns (e.g. *she*, *they*). A character is most often a human agent in a story, although occasionally, it could be some sort of non-human agent, such as a personified animal. In rare cases, groups of individuals or organizations are given the status of character if they consistently operate as a single agent throughout a story (e.g. *militia*).
The detection of all aliases in a given text is a prerequisite to the character resolution task. Once detected, the task proceeds with annotators assigning characters to each alias using four types of tags: CHARACTER, NON-CHARACTER, OTHER, and UNKNOWN. A CHARACTER tag is defined for each character in the story. The NON-CHARACTER tag is a single tag that designates an alias as not referring to a character (e.g. references to people in hypothetical or figurative speech may be identified as aliases but don't refer to characters in the story). The OTHER tag is applied to aliases that already refer to multiple characters, but also refer to other non-character persons in the given story (e.g. *Bennet household* includes members of the *Bennet* family who are characters in Pride and Prejudice, as well as those who are not). Finally, the UNKNOWN tag is assigned to aliases that refer to some character but that character cannot be unambiguous identified.

3. CHARLES

We built CHARLES as a web-based, distributed tool to support multiple annotators (not necessarily co-located) in the character resolution task, simultaneously. In the following sub-sections, we describe the annotation workflow and features supported by CHARLES, the tool's technical design, and then discuss how the tool can be adapted to other annotation tasks.

3.1. Annotation Workflow and Features

Resolving aliases to characters with CHARLES is conducted through a web-interface, enabling platform-independent access to the tool. A story is partitioned into set of individual documents, with aliases pre-identified, and then uploaded to CHARLES' server (this is performed by an administrator), after which each annotator can access those documents she has been assigned. The basic annotation workflow for an annotator is described in the following steps:

1. The annotator logs in and selects the next document to annotate (Documents are segments comprising the whole literary text and annotators essentially proceed through them in chronological order of the story). The document and tagging panel are then presented, the left and right panes in Figure 1, respectively. Annotators can only view one document at a time.

2. The unresolved aliases are marked throughout the document in bright red, as shown in Figure 1. The annotator resolves one alias at a time by selecting it and choosing one or more of the tags (described in Section 2.) displayed in the right panel. Once chosen, the alias is no longer marked as unresolved in the document and is labelled according to the assigned tag(s). If the annotator makes an error, she can easily reverse her actions. For convenience, CHARLES permits annotators to temporary label aliases as *UNRESOLVED* and resolve them once they attain more contextual information. Moreover, annotators can search the CHARACTER tag list to avoid scrolling through long lists.

3. To ease the tagging of plural aliases, annotators can create and assign group tags, which bring selected CHARACTER tags under a single tag (The *Daughters* tag, shown in Figure 1, captures all the *Bennet* daughters in Pride and Prejudice, which are referenced by a plural alias numerous times throughout the text).

4. When the annotator encounters an alias that references a character not yet entered into the tag list, she can create a new CHARACTER tag. The tag is given the name of the underlying alias (adding extract numerical characters if necessary to ensure it is unique, e.g. *Jane (1)* if *Jane* already exists) and can be easily changed if a more suitable name is encountered later. If other annotators are working at the same time, they are instantly notified of the newly created tag and can use it right away for their annotations. If the annotator is unsure of whether a particular alias refers to a new character, CHARLES allows them to easily search already-tagged instances according to given tags (shown in Figure 2). This helps prevent the tag list from becoming corrupted with duplicates and helps track down previously *UNRESOLVED* aliases.

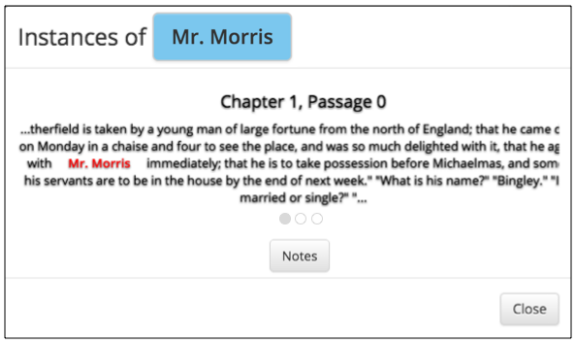

Figure 2: The modal window shown when searching for aliases resolved to a given character, in this case *Mr. Morris*.

We describe a few additional features of the system here:

- If at any point a duplicate CHARACTER tag is entered (a tag corresponding to the same character as another tag), an annotator can easily replace the duplicate tag with another tag, effectively converting all aliases under the duplicate tag to the replacement tag.

- CHARLES supports a number of keyboard shortcuts. For example, annotators can use the keyboard for navigating through aliases in a document, making the activity of annotating more efficient.

- The annotator interface permits document navigation through *previous* and *next* buttons (for navigating to

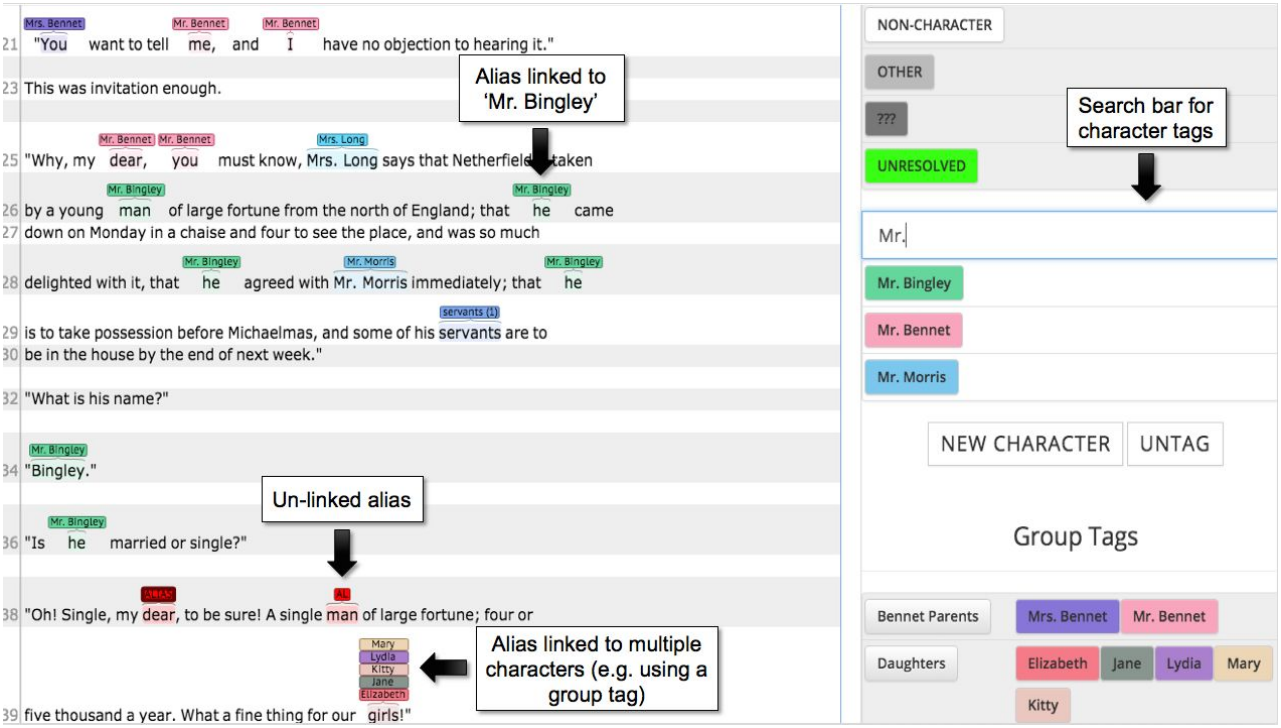

Figure 1: A portion of the CHARLES user interface, showing the document and tagging panel in the left and right panes, respectively. The document shows both tagged and untagged aliases, as well as aliases resolved to single characters and an alias resolved to multiple characters. The panel is comprised of those tags that are not CHARACTER tags, followed by CHARACTER tags, and then group tags. The panel supports a search bar to aid annotators in finding tags easily.

the document immediately before or after, respectively, in the story), as well as through a modal dialog for direct navigation to any document.

- Finally, CHARLES has a built-in chat system so that annotators can collaborate and confer to maintain the consistency of the global character list.

3.2. Technical Design

The front-end component of CHARLES has been developed in HTML, CSS, and Javascript through the AngularJS[2] framework, and employs a variant of the annotation UI offered by brat (Stenetorp et al., 2012) to display the document and tagged instances. The document and annotation data is stored in .json format on the cloud service, Firebase[3], which automatically handles realtime synchronization of the data across users. An additional server is set-up and linked to these components to handle complex querying of the data through a RESTful API.

3.3. Adapting to Other Annotation Tasks

CHARLES is a flexible annotation tool that can support other types of tagging tasks as well, especially those that are cross-document and require a dynamic tag list (e.g. cross-document entity coreference resolution where the entities are not known in advance). The only requirement is that the task conform to assigning labels, from a global list, to pre-detected expressions in text.

The tool offers annotators the ability to singly- or multiply-tag given instances, supporting tag groupings for convenience, and gives annotators control over the global tag list. Furthermore, if the task mandates collaboration among annotators, CHARLES' supports multiple users simultaneously with real-time synchronization of annotations, and a chatting and notification feature for interaction.

CHARLES is released under the Apache Commons 2.0 license and is available on GitHub at the following URL, `https://github.com/hardik-vala/charles`. It is open-source and can be freely downloaded (for installation and customization instructions, visit the provided URL).

4. Pilot Study

The pilot study consisted of manually resolving the characters for all aliases in the novel *Pride and Prejudice*, by Jane Austen, studied extensively in previous works (Ardanuy and Sporleder, 2014; He et al., 2013; Elsner, 2012). The

[2]https://angularjs.org
[3]https://www.firebase.com

annotations were conducted by two annotators, beginning with the alias annotations.

The aliases in Pride and Prejudice were annotated using the brat[4] annotation tool (Stenetorp et al., 2012). The final alias-annotated novel comprised of 17,744 aliases and the accuracy of the annotations, assessed on 7 shared chapters from Pride and Prejudice (out of 52 total) annotated by an expert, are shown in Table 1 (The inter-annotator agreement was 0.974).

Precision	0.987
Recall	0.979
F1-Score	**0.983**

Table 1: Alias annotation accuracy results on 7 chapters from Pride and Prejudice (as compared to an expert annotator).

Once the aliases were identified, we proceeded to resolving them to characters, using CHARLES to facilitate the annotations. The process revealed 144 characters (verified for duplicates) and the accuracy of the character annotations, judged on the same 7 chapters[5] annotated by an expert, are presented in Table 2 (The inter-annotator agreement was 0.972).

	Manual	Automatic
Precision	0.962	0.437
Recall	0.989	0.714
F1-score	**0.975**	0.542

Table 2: Accuracy results for manual character annotation versus an automated method (Stanford's coreference resolution system), on 7 chapters from Pride and Prejudice (as compared to an expert annotator).

Table 2 compares the accuracies of the manual annotation and an automated method, namely the Stanford coreference resolution system (Lee et al., 2011), showing the former yields superior performance. Most errors of the manual approach can be attributed to human error and aliases that are indefinite plural pronouns, such as *Everyone* in the following,

Everyone in the room said how well she looked.

where it is unclear which characters are present in the *room*.

5. Related Works

The brat (Stenetorp et al., 2012) and WebAnno[6] (Yimam et al., 2013) systems are designed to be general-purpose tools and facilitate the manual annotation for a number of common linguistic tasks (including entity recognition, relation extraction, and coreference resolution) through a user-friendly and intuitive browser-based interface. The task of character recognition most closely resembles the NLP task of cross-document entity coreference resolution, but neither brat, nor WebAnno support this task without prior knowledge of all the unique entities (i.e. the tag set for entities must be pre-loaded and is fixed during annotation). In our case, using either system would require all characters be known prior to the resolution of character mentions. CHARLES is more flexible and facilitates the management of a single global list of characters across annotators, while permitting annotators to dynamically add new characters as they encounter them during the annotation process. The tool has built-in measures that ensure the list is duplicate-free and consistent.

CROMER (Girardi et al., 2014) is a web-based annotation tool specifically designed for cross-document entity coreference resolution. Unlike brat and WebAnno, it provides annotators the ability to enter new entities as they see fit. But the workflow is designed such that entity tags are defined prior to the annotation of documents, unlike CHARLES which intertwines the process of annotating documents and defining new characters, resulting in a more seamless annotation experience. Moreover, CROMER requires the user to enter the name for a new entity tag, which may collide with existing names, increasing the likelihood of duplicate entries and confusion among all the annotators. CHARLES automatically generates unique and meaningful tag names. Moreover, it allows users to disambiguate tags using a search feature which displays tagged instances under a particular tag with their surrounding contexts. Finally, CHARLES' built-in chat system allows annotators to confer with each other on the identity of particular tags. These features help keep the character list duplicate-free. (The EDNA plugin for Callisto (Day et al., 2008) is another web tool designed for cross-document entity coreference resolution but is less flexible than CROMER (Girardi et al., 2014).)

The annotation scheme for alias detection and character resolution most closely resembles the annotation guidelines set out by the Automatic Content Extraction program for the entity recognition problem (Doddington et al., 2004). However, characters are a type of entity that require special care in recognizing. They must be distinguished from non-character entities mention in text that may carry an animate signal (e.g. figurative and hypothetical persons), which usually requires a pragmatic understanding of the text.

Finally, there has been recent work in developing automated methods for character detection, but Vala et al. (2015) show they all achieve limited performance, underscoring the need for high-quality, manually annotated corpora generated using comprehensive schemes and efficient annotation tools.

6. Conclusion and Future Work

We have presented a new scheme and annotation tool, CHARLES, for resolving mentions to characters in literary texts and have demonstrated through a pilot study on annotating Pride and Prejudice, that they can be used to produce high-quality results. Furthermore, we release our complete annotations of the work.

An important area of future study is the scalability of CHARLES to larger annotation efforts, involving many an-

[4]http://brat.nlplab.org

[5]These 7 chapters were annotated completely independently.

[6]https://webanno.github.io/webanno/

notators. The pilot study consisted of only two annotators and the system has shown no diminishing performance with up to four simultaneous users, although the upper bound is not yet known. Furthermore, the integrity of the tag list requires the annotators to jointly maintain it, which may not scale with large groups of annotators. The job of preventing duplicate characters from entering the tag list requires manual effort. It is a non-trivial problem not suited yet for automation for two reasons: (1) characters maybe referred to by multiple names, sometimes drastically different (e.g. *Dr. Jeykll* and *Mr. Hyde*). (2) Discerning the identities of mentions referring to minor characters (characters of low relative importance) is difficult, given such characters are not mentioned often in stories and when they are, mentions that almost or exactly match do no necessarily refer to the same character (e.g. there are many servant characters in Pride and Prejudice that play a minor role and each is often referred to using the same mention, *servant*). These problems limit the effectiveness of automated approaches, especially those using string similiarity, and offer area for further work. Some additional modifications planned for CHARLES to better accommodate scale include the incorporation of user roles (e.g. administrator), built-in measures to restrict users from interacting on shared passages intended for calculating agreement (currently, all documents are viewable by all annotators), and a note feature that allows annotators to associate textual notes with each tag to help with tracking.

7. Acknowledgements

Thank you to the annotators for their diligence and patience and to the three reviewers for their helpful comments and suggestions.

8. References

Agarwal, A., Rambow, O., and Passonneau, R. J. (2010). Annotation scheme for social network extraction from text. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 20–28. Association for Computational Linguistics.

Agarwal, A., Corvalan, A., Jensen, J., and Rambow, O. (2012). Social network analysis of alice in wonderland. In *Workshop on Computational Linguistics for Literature*, pages 88–96.

Agarwal, A., Kotalwar, A., and Rambow, O. (2013). Automatic extraction of social networks from literary text: A case study on alice in wonderland. In *the Proceedings of the 6th International Joint Conference on Natural Language Processing*.

Ardanuy, M. C. and Sporleder, C. (2014). Structure-based clustering of novels. In *Proceedings of the EACL Workshop on Computational Linguistics for Literature*, pages 31–39.

Bamman, D., O'Connor, B., and Smith, N. A. (2014a). Learning latent personas of film characters. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, page 352.

Bamman, D., Underwood, T., and Smith, N. A. (2014b). A bayesian mixed effects model of literary character. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 370–379.

Day, D., Hitzeman, J., Wick, M. L., Crouch, K., and Poesio, M. (2008). A corpus for cross-document co-reference. In *LREC*.

Doddington, G. R., Mitchell, A., Przybocki, M. A., Ramshaw, L. A., Strassel, S., and Weischedel, R. M. (2004). The automatic content extraction (ace) program-tasks, data, and evaluation. In *LREC*, volume 2, page 1.

Elsner, M. (2012). Character-based kernels for novelistic plot structure. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 634–644. Association for Computational Linguistics.

Elson, D. K., Dames, N., and McKeown, K. R. (2010). Extracting social networks from literary fiction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 138–147.

Girardi, C., Speranza, M., Sprugnoli, R., and Tonelli, S. (2014). Cromer: A tool for cross-document event and entity coreference. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, may*.

He, H., Barbosa, D., and Kondrak, G. (2013). Identification of speakers in novels. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1312–1320.

Jayannavar, P. A., Agarwal, A., Ju, M., and Rambow, O. (2015). Validating literary theories using automatic social network extraction. In *Proceedings of the NAACL-2015 Workshop on Computational Linguistics for Literature*.

Lee, J. and Yeung, C. Y. (2012). Extracting networks of people and places from literary texts. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*.

Lee, H., Peirsman, Y., Chang, A., Chambers, N., Surdeanu, M., and Jurafsky, D. (2011). Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 28–34. Association for Computational Linguistics.

Moretti, F. (2005). *Graphs, maps, trees: abstract models for a literary history*. Verso.

Sack, G. (2012). Character networks for narrative generation. In *Intelligent Narrative Technologies: Papers from the 2012 AIIDE Workshop, AAAI Technical Report WS-12-14*, pages 38–43.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). Brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. Association for Computational Linguistics.

Vala, H., Jurgens, D., Piper, A., and Ruths, D. (2015). Mr. bennet, his coachman, and the archbishop walk into a bar but only one of them gets recognized: On the difficulty of detecting characters in literary texts. In *Proceedings*

of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 769–774.

Yimam, S. M., Gurevych, I., de Castilho, R. E., and Biemann, C. (2013). Webanno: A flexible, web-based and visually supported system for distributed annotations. In *ACL (Conference System Demonstrations)*, pages 1–6.