Spring 2018

HAIL HYDRA: Named Entity Resolution, Extraction, and Linking of Lexically Similar Names

Cora Schneck
Cora.Schneck@Colorado.EDU

Follow this and additional works at: https://scholar.colorado.edu/honr_theses

Part of the American Literature Commons, Children's and Young Adult Literature Commons, Databases and Information Systems Commons, Digital Humanities Commons, Literature in English, British Isles Commons, Literature in English, North America Commons, and the Other Computer Sciences Commons

Recommended Citation
https://scholar.colorado.edu/honr_theses/1566

This Thesis is brought to you for free and open access by Honors Program at CU Scholar. It has been accepted for inclusion in Undergraduate Honors Theses by an authorized administrator of CU Scholar. For more information, please contact cuscholaradmin@colorado.edu.
Spring 2018

HAIL HYDRA: Named Entity Resolution, Extraction, and Linking of Lexically Similar Names

Cora Schneck

Follow this and additional works at: https://scholar.colorado.edu/honr_theses

Part of the American Literature Commons, Children's and Young Adult Literature Commons, Databases and Information Systems Commons, Digital Humanities Commons, Literature in English, British Isles Commons, Literature in English, North America Commons, and the Other Computer Sciences Commons

This Thesis is brought to you for free and open access by Honors Program at CU Scholar. It has been accepted for inclusion in Undergraduate Honors Theses by an authorized administrator of CU Scholar. For more information, please contact cuscholaradmin@colorado.edu.
HAIL HYDRA

Named Entity Resolution, Extraction, and Linking of Lexically Similar Names

Cora Yanar Schneck

April 6, 2018

Advisor:
Dr. Chenhao Tan, Department of Computer Science

Defense Committee:
Dr. Daniel Jones
Dr. Shivakant Mishra
Dr. Chenhao Tan

Honors Thesis
Computer Science, B.A.
Department of Computer Science
University of Colorado, Boulder
Abstract

Characters and ideas in text are represented by names. A casual reader would have no trouble understanding that a passing reference to Mr. Holmes, Mr. Sherlock Holmes, Sherlock Holmes, and Holmes all trace back to the world’s most famous detective. Names are often shortened or rearranged with common abbreviation or elaborate titles. Each version of a character’s name can be understood as a single head on a multi-headed hydra, all tracing back to the same body. Raw text analysis requires more literary context about how English is structured and how words in a sentence interact to generate the most accurate named entities possible. Many intelligent-dependency parsers and natural language processing systems study text without accounting for how dynamic language can be. This thesis considers the entire body of a piece of literature to identify and relate entities within the same text, regardless of the fluid nature of the exact reference to an entity in literature. Once an entity has been identified, lexically similar names, which refer to the same character, can be linked together to form a global named entity that represents all forms of the named entity referenced in the text. By utilizing raw text as opposed to labeled corpus, this thesis will generate named entities from the text.

Formalizing the problem: What is a name entity?

In computational linguistics, an entity is a generic term, which, for the purposes of this paper, will refer to a distinguishable object in the real or a fictional world (Blessing et al. 2016). This term encompasses named characters (e.g. Mary Lennox), named places (e.g. City of London), or distinct abstract concepts (e.g. Big Brother).

The model (Hydra\(^1\)) designed and developed for this thesis takes raw text files and generates the referenced named entities. The entire process is fully automated, including parsing, identification, and entity linking. There are no pre-defined characters lists, as many similar named entity resolution models that depend on such lists can be biased to the importance of minor characters (Blessing et al. 2016). Instead, the model uses a series of recognizable linguistic features to identify characters from the raw text to build a character list dynamically. The importance of a character is then determined by the frequency of mentions in the text. The model is agnostic to the genre or language

\(^1\)github.com/cyschnack/Hydra
of the raw text, although the current dependency tree parser used in this thesis is optimized for English text. As a result, this model is particularly useful for long text files, common in much of classical literature.

To generalize the model for any type of text, there are a number of pre-processing steps, which this model has automated. Starting from the text, each sentence is tokenized to remove extraneous whitespace, confusing syntax, and inconsistent sentence structures in order to standardized the prose between different sources of literature.

As a reader, there are many familiar elements to sentence structure. In American English, it is common to use a period as a form of internal syntax to indicate common abbreviations (e.g. Dr. abbreviates Doctor and Mr. abbreviates Mister). These types of titles are important in tracking an individual character as Mr. Samsa and Mrs. Samsa are two separate characters. In addition, these types of 'internal' periods can cause the code to prematurely tokenize a sentence (which traditionally end in periods). Once these type of syntactically inconsistent forms of sentence structure have been cleaned from the text, it can be standardized and passed into a part-of-speech tagger.

**SyntaxNet: Part-of-Speech Tagging**

The most computationally tasking step of pre-processing text is translating a sentence into its relevant parts of speech. Currently, the publicly available SyntaxNet, developed by Google Inc., can achieve among the highest levels of accuracy in part-of-speech tagging (94.61%) (Andor et al. 2016). This represents the current upper limit of speech tagging. English is a complicated language to study, so the remaining 5% accuracy is difficult to achieve.

This disparity is caused by several part-of-speech tagging errors, such as: inconsistent labeling in available text, lexicon gaps, and/or unclear/unknown words (Manning 2011). In fact, some of these lexical forms even confuse humans.

For example, an antanaclasis is a form of word-play where a word is repeated within the same phrase and each time the word is used it has a different meaning. This is a favorite rhetorical device of William Shakespeare, and can be seen throughout his considerable breadth of work. Consider Hamlet’s graveside remarks as he ponders the nature of mortality over the skull of an unknown man.
This fellow might be in’s time a great buyer of land, with his statues, his recognizances, his \textit{fines}, his double vouchers, his recoveries. Is this the \textit{fine} of his \textit{fines}, and the recovery of his recoveries, to have \textit{fine} pate full of \textit{fine} dirt?

Hamlet (Act 5, Scene 1, 105-110)

Within two sentences, the word ‘fine’ is used in four different contexts: as a reference to debts, as an older term for deeds, as a description of a ‘good’ skull, and to describe how the dirt feels. Lexical ambiguity is a part of English, and it is often celebrated by authors and poets. They are used for humor\textsuperscript{2} and double entendre (if you know what I mean...). However, these types of wonderful idiosyncrasies in language have stymied language analyzing systems.

In this thesis, the most recent released model from SyntaxNet, Parsey McParseface\textsuperscript{3} was used to tag sentences. Using Parsey, the following sentence is broken apart:

“Many years later, as he faced the firing squad, Colonel Aureliano Buend was to remember that distant afternoon when his father took him to discover ice.”- \textit{One Hundred Years of Solitude} (García)

\textsuperscript{2}Time flies like an arrow, fruit flies like a banana (Pinker 209)

\textsuperscript{3}Full code found at: github.com/tensorflow/models/tree/master/research/syntaxnet
Figure 1: CoNLL SyntaxNet Output

Parsing text takes time, but it scales predictably with size (see figure 5). Parsey was trained on a large volume of different kinds of text, from the Wall Street Journal through the Penn Treebank (Andor et al. 2016). As seen in figure 1, it takes approximately five seconds per sentence for the system to parse and develop a dependency tree as well as save the relevant information in a csv file.
Figure 2: runtime for SyntaxNet
The purpose of proper names in text and society is to specify an individual. A proper noun by itself can be a named entity; for example, a one-word person or place (e.g. Jeeves in *My Man Jeeves*). However, text is normally more sophisticated than a single named entity. Names are universal in human culture, but what specifically defines a name can be difficult to pin down. Many names date back thousands of years, with some of the earliest records dating back to ca. 3100-2900 B.C.
In most Western cultures, a name can be broken into three parts: a given name, a surname, and an optional middle name. However, the order and importance of these sub-entities can vary based on region.

For this thesis, multiple proper nouns will be considered a single entity if they are adjacent in text, for example:

Wilhelm Gottsreich Sigismond von Ormstein (*The Adventures of Sherlock Holmes*)

Sydney Cecil Vivian Montmorency (*A Little Princess*)

Lilian Evangeline Maud Marion (*A Little Princess*)

This can be further expanded to included honorific titles. An honorific title is a title that conveys a specific rank for an individual (e.g. Mr.). These titles can vary in type, frequency of use, and implied formality between cultures. They can be specific to a gender or gender-neutral.

The following titles in Tables 1, 2, 3, and 4 are only a small fraction of all titles used in human society, and when they are used in literature they are often abbreviated and interchanged within the same text (Table 4).

**Table 1**: Common male honorifics used in this code

<table>
<thead>
<tr>
<th>M</th>
<th>Mr.</th>
<th>Sir</th>
<th>Lord</th>
<th>Don</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master</td>
<td>Gentleman</td>
<td>Sire</td>
<td>Esq.</td>
<td>Mester</td>
</tr>
<tr>
<td>Father</td>
<td>Brother</td>
<td>Rev.</td>
<td>Reverend</td>
<td>Brother-in-Law</td>
</tr>
<tr>
<td>Fr.</td>
<td>Pr.</td>
<td>Pastor</td>
<td>Br</td>
<td>Mister</td>
</tr>
<tr>
<td>His</td>
<td>Rabbi</td>
<td>Imam</td>
<td>Sri</td>
<td>Grandfather</td>
</tr>
<tr>
<td>Thiru</td>
<td>Raj</td>
<td>Son</td>
<td>Monsieur</td>
<td>Commodore</td>
</tr>
<tr>
<td>Baron</td>
<td>Prince</td>
<td>King</td>
<td>Emperor</td>
<td>Gentlemen</td>
</tr>
<tr>
<td>Grand Prince</td>
<td>Grand Duke</td>
<td>Duke</td>
<td>Sovereign Prince</td>
<td>Mistah</td>
</tr>
<tr>
<td>Count</td>
<td>Viscount</td>
<td>Crown Prince</td>
<td>Widower</td>
<td>Uncle</td>
</tr>
</tbody>
</table>
Table 2: Common female honorifics used in this code

<table>
<thead>
<tr>
<th>Mrs.</th>
<th>Ms.</th>
<th>Miss</th>
<th>Lady</th>
<th>Mistress</th>
</tr>
</thead>
<tbody>
<tr>
<td>Madam</td>
<td>Ma’am</td>
<td>Dame</td>
<td>Mother</td>
<td>Sister</td>
</tr>
<tr>
<td>Sr.</td>
<td>Her</td>
<td>Kum</td>
<td>Smt.</td>
<td>Ayah</td>
</tr>
<tr>
<td>Daughter</td>
<td>Madame</td>
<td>Mme.</td>
<td>Mademoiselle</td>
<td>Mille.</td>
</tr>
<tr>
<td>Baroness</td>
<td>Maid</td>
<td>Empress</td>
<td>Queen</td>
<td>Archduchess</td>
</tr>
<tr>
<td>Grand Princess</td>
<td>Princess</td>
<td>Duchess</td>
<td>Sovereign Princess</td>
<td>Countess</td>
</tr>
<tr>
<td>Gentlewoman</td>
<td>Aunt</td>
<td>Widow</td>
<td>Doha</td>
<td>Comtesse</td>
</tr>
<tr>
<td>Baronne</td>
<td>Grandmother</td>
<td>Sister-in-Law</td>
<td>Missus</td>
<td>Headmistress</td>
</tr>
</tbody>
</table>

Table 3: Common neutral honorifics used in this code

<table>
<thead>
<tr>
<th>Dr.</th>
<th>Doctor</th>
<th>Captain</th>
<th>Capt.</th>
<th>Professor</th>
<th>old</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prof.</td>
<td>Hon.</td>
<td>Honor</td>
<td>Excellency</td>
<td>Honourable</td>
<td>silly</td>
</tr>
<tr>
<td>Honorable</td>
<td>Chancellor</td>
<td>Vice-Chancellor</td>
<td>President</td>
<td>Vice-President</td>
<td>Poor</td>
</tr>
<tr>
<td>Senator</td>
<td>Prime</td>
<td>Minster</td>
<td>Principal</td>
<td>Warden</td>
<td>Cuz</td>
</tr>
<tr>
<td>Dean</td>
<td>Regent</td>
<td>Rector</td>
<td>Director</td>
<td>Mayor</td>
<td>Highness</td>
</tr>
<tr>
<td>Judge</td>
<td>Cousin</td>
<td>Archbishop</td>
<td>General</td>
<td>Secretary</td>
<td>St.</td>
</tr>
<tr>
<td>Saint</td>
<td>San</td>
<td>Assistant</td>
<td>Director</td>
<td>The Right Honorable</td>
<td>The Right Honourable</td>
</tr>
</tbody>
</table>
Table 4: Variations on Titles

<table>
<thead>
<tr>
<th></th>
<th>M.</th>
<th>Mr.</th>
<th>Mister</th>
<th>Mistah</th>
<th>Mester</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ms.</td>
<td>Miss</td>
<td>Missus</td>
<td>‘Mademoiselle’</td>
<td>Mlle.</td>
<td></td>
</tr>
<tr>
<td>Madam</td>
<td>Ma’am</td>
<td>Madame</td>
<td>Mme.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dr.</td>
<td>Doctor</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Capt.</td>
<td>Captain</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prof.</td>
<td>Professor</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>St.</td>
<td>Saint</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cuz</td>
<td>Cousin</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In many Western cultures, it is common for a woman to take their husband’s surname, so by using titles, the parser can better distinguish between characters with the same last name (through marriage or familial ties). For example, “Miss” for an unmarried woman will be used with her original surname while “Mrs.” will have her husband’s. In addition, understanding that there is no difference between the titles “Mr.,” “Mistah,” and “Mister” ensures that they are grouped together as the same entity.

To illustrate the importance of this distinction, there is no different between Mr. Holmes and Mister Holmes, but Mr. Darling and Mrs. Darling describe two entirely different characters.

The final step to recognize named entities in text involves finding the most common words used around proper noun pairings. The most useful of these being: ‘the’ (a definite article) and ‘of’ (preposition). These common connecting words allow for named entities’ proper nouns like “City of London” and “United States of America” to be fully identified in text rather than in piecemeal.

The full titles are usually the longest version of a character name and are used to identify all subsequent versions of a name. Some examples of these extended titles are listed in Table 5.
Table 5: Full Titles from Text

<table>
<thead>
<tr>
<th>Full Title</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black Avenger of the Spanish Main</td>
<td><em>The Adventures of Tom Sawyer</em></td>
</tr>
<tr>
<td>Captain Rollo Bickersteth of the Coldstream</td>
<td><em>My Man Jeeves</em></td>
</tr>
<tr>
<td>Serene Highness the Prince of Saxburg-Leignitz</td>
<td><em>My Man Jeeves</em></td>
</tr>
<tr>
<td>Bagheera of the Council Rock</td>
<td><em>The Jungle Book</em></td>
</tr>
<tr>
<td>Museum of the Faculty of Medicine of Paris</td>
<td><em>20,000 Leagues Under the Sea</em></td>
</tr>
<tr>
<td>Wicked Witch of the West</td>
<td><em>The Wonderful Wizard of Oz</em></td>
</tr>
<tr>
<td>League of the Red-headed Men</td>
<td><em>Sherlock Holmes</em></td>
</tr>
</tbody>
</table>

From the limited dataset, the longest full named entities identified by this model were:

General Manager of the River Company of the Caribbean
Superior of the Academy of the Presentation of the Blessed Virgin

Both of these named entities are from Nobel Laureate Gabriel García Márquez’s novel *Love in the Time of Cholera* published in 1988.

These characters were identified in the following text by the steps illustrated:

(i) Identify proper nouns

<Superior>_<n0> of the <Academy>_<n1> of the <Presentation>_<n2> of the <Blessed>_<n3> <Virgin>_<n4>

(ii) Group adjacent proper nouns

<Superior>_<n0> of the <Academy>_<n1> of the <Presentation>_<n2> of the <Blessed Virgin>_<n3>

(iii) Include connecting words “of” and “the”

<Superior of the Academy of the Presentation of the Blessed Virgin>_<n0>

The final form of a named entity that this code will identify can be written as:

(TITLE) + (PROPER NOUN) + OPTIONAL (CONNECTING WORDS/PROPER NOUN)
Each title and proper noun and connecting word will be merged *ad infinitum* until an element not defined as a title, proper noun, or connecting word breaks the cycle.

‘The Superior of the Academy of the Presentation of the Blessed Virgin’ refers to a single person, specifically an important member at the academy. The additional descriptive words and connecting words distinguish this person at this particular academy. Any further reference to this person could use only a portion of their full title, so it is important to identify the longest possible version of a name in each instance it appears.
In this thesis, the script identifies the named entities and tags it within the text. Nouns and proper nouns are separately identified and counted up from zero (e.g. for nouns: n0, n1, n2, etc. For pronouns: p0, p1, p2, etc.). For an in-line example, consider:

One morning, when <Gregor Samsa> n0 woke from troubled dreams, <he> p0 found <himself> p1 transformed in <his> p2 bed into a horrible vermin. (Metamorphosis)

Unlike google's parser, the runtime of this tagging depends on both the size of the text and the size of the named entity.
Figure 5: Runtime for manually tagging text

However, as a result it reduces the apparent proper nouns found because they were merged into a named entity.
Figure 6: Full word named entities found in text
Figure 7: Full word name entities found in text where the size of the named entity found is independent to the size of the text.

Once all named entities have been identified within the text, they are grouped and linked by similarity to create a global named entity. By clustering together all named entities, all iterations of a name are recursively generated based on the available names\textsuperscript{4}.

For example:

\textsuperscript{4}for more example of named entity trees, see supplement
Once the final global tree has been generated, the tagged text is read through and each reference to either the root of the tree, or to any of its children is summed together and included as a reference to the same character.

The recursive search and linking to create the named entity tree searches for instances of a word that shares at least one element in each sub-tree (e.g. linking Dr. Jekyll and Dr. Henry Jekyll). As seen in the Supplement (re: “Doha Fermina Daza”), this allows for a character to have small changes to their name through the course of the novel, e.g. as Fermina spends most of the novel *Love in the Time of Cholera* married with the surname ’Daza’. However, before her marriage, her surname is ’Sanchez’, and this is included in her named entity tree since that level of the sub-tree searches for ’Fermina’, and finds both ’Fermina Daza’ and ’Fermina Sanchez’.
Texts do not have main characters, so much as “characters of interest”. Many of Jules Verne’s stories are either first person or have a character that acts in place of a narrator that follows around the more important character of interest. In *20,000 Leagues Under the Sea*, Professor Pierre Aronnax might be the narrator, but Captain Nemo is the character of interest. Dr. Watson narrates and records their adventures, but the titular Sherlock Holmes is the character of interest. If they are different genders, this important difference between the narrator and the ‘main character’ becomes even more prominent as the character of interest’s pronoun tends to dominate the text and not the narrator’s.

Once all versions of a name in the text are accounted for, the characters of interest can be identified through their frequency in the text. Logically, a more frequently mentioned character enjoys more of the story’s focus. This disparity is further magnified in first-person narratives where the narrator’s name is infrequently used outside of dialogue.

Some examples are provided in Table 6. \(^5\)

---

\(^5\)for complete list, see the supplement
As seen in several previous examples, named entities do not have to refer to a character in a traditional sense. The structure of a story sets man against man just as easily as it pits man against nature, man against society, and man against self. A named entity can be a person or even an idea. For example, in *1984*, the omnipresent Big Brother does not refer to a single person but rather the concept of the totalitarianism that Winston Smith struggles and eventually succumbs under the weight of.
To determine the gender of a given global named entity, the addition of a trained gender classifier is applied to each global entity tree. Using features from a name, the model can determine the sex of a character with 84.075% accuracy from a training corpus of 97,050 names. Additional accuracy is unlikely because many names are interchangeable between genders and difference languages and cultures have different structures for gendered names. Additional methods of gender identification have also been applied based on context. If a gendered title is used, it is confirmation of a given gender regardless of the name. For an example, consider the final result for the name Atticus for the character Attics Finch from To Kill a Mockingbird:

*The name Atticus is most likely Male*

*Odds: Female (0.3), Male (0.7)*

One of the strength of this model is that it can classify a gender to a named entity regardless of the name existing in the corpus of names provided or even in a fictional world. This is particular useful in fantasy and science fiction novels and keeps Hydra genre agnostic as well.

For example, consider the results for the male elf from the fantasy series The Adventure Zone and princess in the science fiction novel The Princess of Mars:

*The name Taako is most likely Male*

*Odds: Female (0.490909090909), Male (0.509090909091)*

*The name Dejah is most likely Female*

*Odds: Female (0.852272727273), Male (0.147727272727)*

To increase gendering accuracy, some more features of the naming structure have to take be taken into account. For example, the last name of a character does not determine the gender, however, it needs be considered, especially for characters that are not given any additional names or gendered titles.

For simple example, consider the character of Gregor Samsa from Franz Kafkas Metomorphosis:
Pre-Weighting

Gregor Samsa (Female: 2.42291 Male: 1.57709)

- Gregor
  - Gregor
    - Female: 0.29787
    - Male: 0.70213
  - Gregor Samsa
    - Female: 1.21145
    - Male: 0.78855
- Samsa
  - Samsa
    - Female: 0.91358
    - Male: 0.08642
Post-Weighting

Gregor Samsa (Female: 1.143888  Male: 1.456112)
  └ Gregor
    └ Gregor
       └ Female: 0.29787
       └ Male: 0.70213
    └ Gregor Samsa
       └ Female: 0.571944
       └ Male: 0.728056
  └ Samsa
    └ Samsa
       └ Female: 0.274074
       └ Male: 0.025926

For a more complex example, consider:
Applications in Dynamic Relationships

By using sentiment analysis, a subjective step in the utilization of the same global entities tree can establish dynamic relationships through text. This can be done with any text sentiment analyzer. For this purpose of a demo, this script uses the NLTK sentiment analyzer.

The original raw text is first tokenized into sections and each section is passed to NLTK to determine the relative sentiment of a given section of text. The sentiment is then tracked based on character interactions. A character can be considered interacting with another if they are mentioned in the same section that has been tokenized. For the purposes of testing, the number of sentences in each section is eight, to approximate the typical paragraph size. In smaller text, this is changed to three, to account for the decrease in overall size and to find small instances of interaction.

Figure 8: Metamorphosis’s network of interaction where red is positive sentiment and blue is negative sentiment.

23
Applications in Gendered Sentiment

The narrative progression of a fictional novel is best expressed by Campbell’s *The Hero With a Thousand Faces*:

“A hero ventures forth from the world of the common day into a region of supernatural wonder: fabulous forces are there encountered and a decisive victor is won: the hero comes back from this mysterious adventure with the power to bestow boons on his fellow man.” (Campbell 28)

So, with the gender and global named entities derived, it is possible to observe the sentiment applied to a particular gender over the course of a book and to discover if it varies between sexes. Because the datasets I have used in this thesis are mostly classical texts, only a small sub-set of them include both prominent female and male characters. However, within that subset, two show a statistically significant difference in the sentiment between men and women through the course of the novel: *A Princess of Mars* and *A Scarlet Letter*.

<table>
<thead>
<tr>
<th>Title</th>
<th>Difference?</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peter Pan</td>
<td>No</td>
<td>0.4924</td>
</tr>
<tr>
<td>The Wonderful Wizard of Oz</td>
<td>No</td>
<td>0.3284</td>
</tr>
<tr>
<td>Secret Garden</td>
<td>No</td>
<td>0.5949</td>
</tr>
<tr>
<td>A Princess of Mars</td>
<td>Yes</td>
<td>$2.1 \times 10^{-5}$</td>
</tr>
<tr>
<td>The Scarlet Letter</td>
<td>Yes</td>
<td>$7.32 \times 10^{-6}$</td>
</tr>
</tbody>
</table>
A Princess of Mars

This classic pulp science-fiction novel, written in 1912 by Edgar Rice Burroughs, follows John Carter, a Confederate soldier who finds himself mysteriously transported to Mars. The story focuses on his adventures with the Martian natives and his love interest, the lovely Princess Dejah Thoris. While it would require a more in-depth research and analysis of the novel, the original cover and the genre lend itself to a traditional “damsel-in-distress” plot line. Because Dejah is one of the few women in the novel, her overall sentiment is largely impacted by the series of kidnapping and rescues she endures.
Figure 9: Princess of Mars
Figure 10: Princess of Mars for Women
Figure 11: *Princess of Mars* for Men
As the full title suggests, *The Scarlet Letter* can be considered a romance. However many of the main themes of the novel, written by Nathaniel Hawthorne in 1850, surround sin and societal stigma as the main character of the novel, Hester Prynne, gives birth to a baby girl as the result of a secret affair. This book is a prime candidate to see how men and woman behave differently through a novel since the story follows Hester as she is alienated from the community and raises her daughter alone. As a result, the two prominent female characters in the novel, Hester and her daughter Pearl, will experience the brunt of the negative attention and sentiment throughout the novel, damping the moments of positive sentiment and keeping the average sentiment lower than than her male counterparts.
Figure 12: Scarlet Letter
Figure 13: Scarlet Letter for Women
Figure 14: Scarlet Letter for Men
Flexibility of the Architecture and Use in Multi-Language Text

The global named entity trees are decoupled from any particular parser, gender classifier, or sentiment analyses. With an improvement in accuracy to any dependent structure, the global name entities would also have a marked improvement. Parsey is a powerful dependent tree parser, however, it can mislabel unfamiliar words or words in an unfamiliar dialect. The code developed for this thesis has the ability to remove specific offending words (see Table 8) while still preventing the model from becoming overfit to any particular style of writing.

<table>
<thead>
<tr>
<th>Table 8: Words to Ignore</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chapter</td>
</tr>
<tr>
<td>Volume</td>
</tr>
<tr>
<td>O</td>
</tr>
<tr>
<td>Anon</td>
</tr>
<tr>
<td>Ought</td>
</tr>
<tr>
<td>Thou</td>
</tr>
<tr>
<td>Thither</td>
</tr>
<tr>
<td>Yo</td>
</tr>
<tr>
<td>Till</td>
</tr>
<tr>
<td>Ay</td>
</tr>
<tr>
<td>Hitherto</td>
</tr>
<tr>
<td>Ahoy</td>
</tr>
<tr>
<td>Alas</td>
</tr>
<tr>
<td>Thy</td>
</tr>
<tr>
<td>Thee</td>
</tr>
<tr>
<td>Good-Night</td>
</tr>
<tr>
<td>Good-morning</td>
</tr>
<tr>
<td>To-day</td>
</tr>
<tr>
<td>To-Morrow</td>
</tr>
<tr>
<td>Tis</td>
</tr>
<tr>
<td>Good-Will</td>
</tr>
<tr>
<td>To-Day</td>
</tr>
<tr>
<td>D’You</td>
</tr>
<tr>
<td>O’er</td>
</tr>
<tr>
<td>Aye</td>
</tr>
<tr>
<td>Beheld</td>
</tr>
<tr>
<td>Nay</td>
</tr>
<tr>
<td>So-And-So</td>
</tr>
<tr>
<td>Thereupon</td>
</tr>
<tr>
<td>Twas</td>
</tr>
<tr>
<td>Tha</td>
</tr>
<tr>
<td>Tha’Rt</td>
</tr>
<tr>
<td>Eh</td>
</tr>
<tr>
<td>Wither</td>
</tr>
<tr>
<td>Ah</td>
</tr>
<tr>
<td>Methought</td>
</tr>
<tr>
<td>Wilt</td>
</tr>
<tr>
<td>Wherefore</td>
</tr>
<tr>
<td>Doth</td>
</tr>
<tr>
<td>Betwixt</td>
</tr>
<tr>
<td>Dat</td>
</tr>
<tr>
<td>Withal</td>
</tr>
<tr>
<td>Thyself</td>
</tr>
<tr>
<td>Sayeth</td>
</tr>
<tr>
<td>Spake</td>
</tr>
</tbody>
</table>

Evaluation and Discussion

This paper presents a model to identify and group lexically similar named entities into a single global entity that represents the entire class. The generalized application of this model can find all instances of a named entity with the specified features (e.g. titles, proper nouns, and connecting words). The current model finds 4.32% false positives, typically struggling with antiquated phrases that it was unlikely trained on (e.g. ‘Nevermore’, ‘Quoth’). In the future, this can be accounted for by adding common offenders to the ‘Words to Ignore’ (Table 8). As expected, because SyntaxNet is trained on a corpus of words, it had trouble finding and labeling initials. For example, the main
character in *The Trail* is named ‘K’, and it did not find his ‘name’. The model also misses uncommon nicknames. As a result, the false negatives of this model is 12.082%, which can be improved by a specialized code for finding these commonly missed names and/or a new part-of-speech tagger. However, this model ultimately succeeds in its ability to compress multiple instances of the name in different forms to allow them to be linked and condensed into a single named entity for networking and reference opportunities.

**References**


“Cuneiform Tablet: Administrative Account of Barley Distribution with Cylinder Seal Impression of a Male Figure, Hunting Dogs, and Boars.” *The Met’s Heilbrunn Timeline of Art History*, www.metmuseum.org/toah/works-of-art/1988.433.1/.


Names Corpus, Version 1.3 (1994-03-29) Copyright (C) 1991 Mark Kantrowitz Additions by Bill Ross.


Shakespeare, William. Hamlet, Prince of Denmark.*Project Gutenberg*, 1 Nov. 1998,
Gutenberg.org Datasets:
Burnett, Frances Hodgson. “A Little Princess” Project Gutenberg, 1 July 1994, gutenberg.org/ebooks/146.

Full Code, raw text, and trained models available at: github.com/cyschneck/Hydra
Supplement

More Named Entity Trees

Miss Roylott

- Miss
  - Miss Roylott

Roylott

- Roylott
  - Roylott of Stoke Moran

Red-headed League

- Red-headed
  - Red-headed Men
  - Red-headed League

City of London

- City
  - City of London
  - London
    - London Road
    - London Bridge

Dr. Grimesby Roylott

- Dr.
  - Dr. Roylott
  - Dr. Grimesby Roylott

Grimesby

- Grimesby
  - Grimesby Roylott

Roylott

- Roylott
  - Roylott of Stoke Moran
Ministry of Plenty
- Ministry
- Ministry of Peace
- Ministry of War
- Ministry of Plenty
- Ministry of Love
- Ministry of Truth
- Plenty
  - Plenty

Ministry of War
- Ministry
- Ministry of Peace
- Ministry of War
- Ministry of Plenty
- Ministry of Love
- Ministry of Truth
- War
  - War

Ministry of Truth
- Ministry
- Ministry of Peace
- Ministry of War
- Ministry of Plenty
- Ministry of Love
- Ministry of Truth
- Truth
  - Truth

Winston Smith
- Winston
- Winston Smith
- Smith
  - Smith
  - Smith W.
Monsieur Aronnax
  | Monsieur
  | Monsieur Aronnax
  | Monsieur Pierre Aronnax
  | Mr. Aronnax
  | Mr. Pierre Aronnax
  | Aronnax
  | Aronnax

Honourable Mr. Batulcar
  | Honourable
  | Honourable Mr. Batulcar
  | Honourable William Batulcar
  | Mr.
  | Mr.
  | Mr. Batulcar
  | Batulcar
  | Batulcar

Mr. Phileas Fogg
  | Mr.
  | Mr.
  | Mr. Fogg
  | Mr. Phileas Fogg
  | Monsieur Fogg
  | Monsieur Phileas Fogg
  | Phileas
  | Phileas
  | Phileas Fogg
  | Fogg
  | Fogg

52
More Characters of Interest (COI) in Text

Note: This is how the script outputs information on Characters of Interest and Gendering (as seen in Table 6)

Princess of Mars (Burroughs)
IS FIRST PERSON TEXT: True
CHARACTER OF INTEREST: [('Princess Dejah Thoris', 171)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Tars Tarkas the Thark', 132), ('Sola', 118), ('Tharks of Barsoom', 113), ('throng of Martians', 107), ('Mors Kajak of Helium', 104)]

Tarzan of the Apes (Burroughs)
IS FIRST PERSON TEXT: False
Predicted gender of main character is 'Male' [('he', 1493)]: True
CHARACTER OF INTEREST: [('Tarzan of the Dum-Dum', 604)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Mr William Cecil Clayton', 286), ('Lieutenant Charpentier', 195), ('Jane Porter', 168), ('Mr Philander', 100), ('Professor Porter', 95)]

Heart of Darkness (Conrad)
IS FIRST PERSON TEXT: True
CHARACTER OF INTEREST: [('Mistah Kurtz', 116)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Company', 13), ('Charlie Marlow', 11), ('English', 10), ('Europe', 8), ('Russian', 7)]

A Christmas Carol (Dickens)
IS FIRST PERSON TEXT: False
CHARACTER OF INTEREST: [('Scrooge the Baleful', 329)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Ghost of Christmas Present', 175), ('Evil Spirit', 79), ('Bob Cratchit', 51), ('Tiny Tim', 22), ('Master Peter Cratchit', 19)]

The Hound of the Baskervillies (Doyle)
The Sign of Four (Doyle)
IS FIRST PERSON TEXT: True
CHARACTER OF INTEREST: [('Mr Sherlock Holmes', 118)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Mr Bartholomew Sholto', 72), ('Miss Mary Morstan', 37), ('Mr Athelney Jones', 33), ('Strange Story of Jonathan Small', 33), ('Toby', 26)]

The Scarlet Letter (Hawthorne)
IS FIRST PERSON TEXT: False
Predicted gender of main character is 'Female' [('her', 934)]: True
CHARACTER OF INTEREST: [('Madame Hester', 293)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Little Pearl', 202), ('Mr Dimmesdale', 71), ('Old Roger Chillingworth', 56), ('Reverend Master Dimmesdale', 54), ('New England Clergyman', 39)]

Metamorphosis (Kafka)
IS FIRST PERSON TEXT: False
Predicted gender of main character is 'Male' [('his', 524)]: True
CHARACTER OF INTEREST: [('Gregor Samsa', 296)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Grete', 25), ('Mr Samsa', 21), ('Mrs Samsa', 10), ('God', 7), ('Christmas', 3)]

The Call of the Wild (London)
IS FIRST PERSON TEXT: False
Predicted gender of main character is 'Male' [('he', 616)]: True
CHARACTER OF INTEREST: [('Buck', 358)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('John Thornton', 102), ('Spitz', 60), ('Francois', 60), ('Perrault', 39), ('Hal', 37)]

White Fang (London)
IS FIRST PERSON TEXT: False
Predicted gender of main character is 'Male' [(he', 1531)]: True
CHARACTER OF INTEREST: [('White Fang', 569)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Grey Beaver', 119), ('Henry', 95), ('Beauty Smith', 84), ('Matt', 82), ('Bill', 81)]

The Strange Case of Dr. Jekyll and Mr. Hyde (Stevenson)
IS FIRST PERSON TEXT: True
CHARACTER OF INTEREST: [('Mr Utterson of Gaunt Street', 126)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Dr Henry Jekyll', 62), ('Edward Hyde', 60), ('Poole', 58), ('Dr Lanyon', 30)]

Treasure Island (Stevenson)
IS FIRST PERSON TEXT: True
CHARACTER OF INTEREST: [('Long John Silver', 295)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Jim Hawkins', 96), ('Doctor Livesey', 63), ('Admiral Benbow', 52), ('Captain Flint', 50), ('Tom Redruth', 46)]

20,000 Leagues Under the Sea (Verne)
IS FIRST PERSON TEXT: True
CHARACTER OF INTEREST: ('Captain Nemo', 283)
TOP CHARACTERS OF INTEREST: [('Commander of the Nautilus', 195), ('Conseil', 193), (Mr Ned Land, 144), ('Captain Denham of the Herald', 126)]

The Island of Doctor Moreau (Wells)
IS FIRST PERSON TEXT: True
CHARACTER OF INTEREST: [('Montgomery', 202)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Island of Doctor Moreau', 136), ('Beast People', 74), ('Sayer of the Law', 61), ("M'ling", 40)]

Time Machine (Wells)
IS FIRST PERSON TEXT: True
CHARACTER OF INTEREST: [('Time Traveller', 104)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('"Little Weena", 49), ('Advancement of Mankind', 28), ('Psychologist', 22), ('moon', 19), ('Editor', 19)]

My Man Jeeves (Wodehouse)
IS FIRST PERSON TEXT: True
CHARACTER OF INTEREST: [('Jeeves', 233)]
ADDITIONAL TOP CHARACTERS OF INTEREST: [('Mr Blooming Lattaker', 101), ('Rocky Todd', 60), ('Old Bicky', 57), ('Bobbie Cardew', 56), ('Corky', 53)]