Automatic Attribution of Quoted Speech in Literary Narrative

David K. Elson and **Kathleen R. McKeown**
Columbia University
{delson, kathy}@cs.columbia.edu

Abstract

We describe a method for identifying the speakers of quoted speech in natural-language textual stories. We have assembled a corpus of more than 3,000 quotations, whose speakers (if any) are manually identified, from a collection of 19th and 20th century literature by six authors. Using rule-based and statistical learning, our method identifies candidate characters, determines their genders, and attributes each quote to the most likely speaker. We divide the quotes into syntactic classes in order to leverage common discourse patterns, which enable rapid attribution for many quotes. We apply learning algorithms to the remainder and achieve an overall accuracy of 83%.

Introduction

Stories are the currency with which we exchange information about our lives. From news to nonfiction tomes to personal blogs, many genres of text on and off the Web use narrative structure as a means for socially conveying information. An understanding of the semantics of storytelling allows us to better reason about the people, events, interactions, and themes found inside volumes of literature, news and other media. In particular, understanding direct and indirect speech is important for tasks such as opinion mining (Balahur et al. 2009), social network extraction (Bird et al. 2006), discourse (Redeker and Egg 2006) and even automatic visualization of a scene (Salesin 1996). In this paper, we address the problem of attributing instances of quoted speech to their respective speakers in narrative discourse – in particular, the domain of English-language literature (including translated works by Russian and French authors).

We compiled works by Chekhov, Flaubert, Twain, Austen, Dickens and Conan Doyle that appeared between 1815 and 1899. Each author was influential in popularizing the form of the novel (or, in the cases of Chekhov and Conan Doyle, the short story) as a medium distinct from the more well-established play or poem. However, these works still hearken back to the older form, in that, like a play, they consist of extended scenes of dialogue between two or more individuals in a scene. These texts have a large proportion of quoted speech (dialogue and internal monologue). Understanding what these texts are about is predicated on identifying the characters in each scene, and what they are saying or thinking.

The baseline approach to this task is to find named entities near the quote and assign the quote to the one that is closest (especially if there is a speech verb nearby). However, even in the straightforward prose by these authors (compared to that of modernist authors), in many instances there is a large distance between the quote and its speaker. For example, in the following passage from Austen's *Emma*, there are several named entities near the quote, and correct attribution depends on an understanding of syntax and (to a lesser extent) the semantics of the scene:

> "Take it," said **Emma**, smiling, and pushing the paper towards **Harriet**– *"it is for you. Take your own."*

The quote "it is for you. Take your own" is preceded by two proper names in the paragraph, Emma and Harriet, of which the correct speaker is the farther of the two. In other cases, such as extended conversations, the quoted speech and the nearest mention of its speaker may be separated by 15, 20 or an even greater number of paragraphs.

In the following sections, we describe a method for achieving two goals: identifying the characters present in a text, and attaching each instance of quoted speech to the appropriate character (if any). Our results show that we can correctly assign a quote to its character – a named entity or nominal we have extracted from the text – for 83% of the quotes by these authors. This is a significant improvement over the baseline.

Related Work

The pragmatics of quoted and indirect speech in literature have long been studied (Voloshinov 1971; Banfield 1982), but the application of natural language processing to literature is limited by comparison; most work in quoted speech identification and attribution has been focused on the news domain. Most recently, Sarmento and Nunes (2009) presented a system for extracting and indexing quotes from online news feeds. Their system assumes that quotes fall into one of 19 variations of the expected syntactic construction "*[Name] [Speech Act] [Quote]*" where *Speech Act* is one of

Author	Title	Year	# Quotes	% Quote	Quotes attributed	Unique speakers	% named
Jane Austen	*Emma**	1815	549	51%	546	36	39%
Charles Dickens	*A Christmas Carol*	1843	495	26%	491	108	10%
Gustave Flaubert	*Madame Bovary**	1856	514	19%	488	126	25%
Mark Twain	*The Adventures of Tom Sawyer**	1876	539	27%	478	55	36%
Sir Arthur Conan Doyle	"The Red-Headed League" "A Case of Identity" "The Boscombe Valley Mystery" "A Scandal in Bohemia"	1890 1888 1888 1888	524	71%	519	40	13%
Anton Chekhov	"The Steppe" "The Lady with the Dog" "The Black Monk"	1888 1899 1894	555	28%	542	61	21%

Table 1: Breakdown of the quoted speech usage in six annotated texts. * indicates that excerpts were used.

35 selected verbs and *Name* is a full mention (anaphoric references are not allowed). Pouliquen et al. (2007) take a similar approach in their news aggregator, identifying both universal and language-specific templates for newswire quotes against which online feeds are matched. This method trades off recall for precision, since there are many syntactic forms a quote may take. Unfortunately, the tradeoff is not as favorable for literary narrative, which is less structured than news text in terms of attributing quoted speech. For example, a quote often appears by itself in a paragraph. Our approach augments the template approach with a supplementary method based on statistical learning.

The work targeting literature has covered character and point-of-view identification (Wiebe 1990) as well as quoted speech attribution in the domain of children's literature for purposes of building a text-to-speech system (Zhang, Black, and Sproat 2003). Mamede and Chaleira (2004) work with a set Portuguese children's stories in their heavily rule-based approach to this task; we aim to be less reliant on rules for processing a larger corpus. Glass and Bangay (2007) focus on finding the link between the quote, its speech verb and the verb's agent. Compared to this work, we focus more on breadth (recall), as we include in our evaluation quotes that do not have speech verbs nearby.

Corpus and its annotation

We selected works by six authors who published in the 19th century for inclusion in our study (see Table 1). The variety is meant to prevent overfitting to the style of any particular author: four authors wrote in English, one in Russian (translated by Constance Garnett) and one in French (translated by Eleanor Marx Aveling); two authors contribute short stories and the rest novels (while Dickens often wrote in serial form, *A Christmas Carol* was published as a single novella). Excerpts were taken from *Emma*, *Madame Bovary* and *The Adventures of Tom Sawyer*.

The full corpus consists of about 111,000 words including 3,176 instances of quoted speech (where *quoted speech* is a block of text within a paragraph falling between quotation marks). To obtain gold-standard annotations of which characters were speaking or thinking which quotes, we conducted an online survey via Amazon's Mechanical Turk program. For each quote, we asked 3 annotators to independently choose a speaker from the list of contextual candidates – or, choose "spoken by an unlisted character" if the answer was not available, or "not spoken by any character" for non-dialogue cases such as sneer quotes. We describe below the method with which we extract candidate speakers, including named entities and nominals, from the text. Up to 15 candidate speakers were presented for each quote from up to 10 paragraphs preceding the paragraph with the quote (including the quote's paragraph itself).

When two definite noun phrases referred to the same person (e.g., "Harriet" and "Emma's friend"), annotators were instructed to choose the reference "most strongly associated" with the quote in question. We did not attempt to develop tools that could determine when such definite noun phrases were coreferent. We initially experimented with a named entity extraction and coreference resolution system called Jet (Grishman, Westbrook, and Meyers 2005); we found that although it could pull proper nouns from the text, it often did not find nominals which we wanted identified (e.g., "her father") and coreference for proper nouns was imprecise (such as from linking opposite genders). Jet had been trained for news (according to the ACE guidelines) rather than literature. For this reason, we developed our own tool for identifying character nominals.

Of the 3,578 quotes in the survey results, 2,334 (about 65%) had unanimous agreement as to the identity of the speaker, and 1,081 (another 30%) had a 2-vote majority which was assumed to be the correct answer. The remaining 4.5% had a total 3-way tie, often in cases where multiple coreferents were offered for the same speaker. We excluded these cases from our corpus, as coreference is not our main focus. To normalize for poor annotator performance, each annotator was graded according to the rate at which he agreed with the majority. If this rate fell below 50%, we threw out all the annotator's ratings; this affected only 2.6% of the votes. We also excluded from evaluation the 239 quotes (7%) where a majority agreed that the correct speaker was not among the options listed (including 3% where the correct character was not chunked, and 4% where the passage did not extend far back enough to determine the speaker). Annotators also agreed that 112 of the quotes (3.5%) were non-dialogue text. We set out to detect such cases alongside quotes with speakers.

We put aside one-third of the corpus for use in developing our method, and left the remainder for training and testing. We have publicly released these data to encourage further work.[1] Table 1 gives the number of quotes for each text as well as the proportion of words in each text that are within quotes. The Sherlock Holmes detective stories, by Conan Doyle, are 71% in quotes on average, where Flaubert's *Madame Bovary* is only 19% in quotes. The latter two columns show the number of unique speakers we identified (where named entities are counted once, and nominals individually); the last column gives the proportion of named characters as opposed to nominals. These suggest differences in the texts' social networks: *Emma* features a small, tight-knit community, where Dickens writes of a more diffuse network.

Methodology

Our method for quoted speech attribution is as follows:

1. **Preprocessing:** We identify all named entities and nominals that appear in the passage of text preceding the quote in question. These are the *candidate* speakers, and for building the statistical models, they match the candidates provided to our annotators. We replace certain spans of text with symbols, and clean or normalize other parts.
2. **Classification.** The second step is to classify the quote into one of a set of syntactic categories. This serves to cluster together scenarios where the syntax strongly implies a particular solution. In some cases, we choose a candidate solely based on its syntactic category.
3. **Learning.** The final step is to extract a feature vector from the passage and send it to a trained model specific to its syntactic category. There are actually n vectors compiled, one for each candidate speaker, that are considered individually. The model predicts the probability that each candidate is a *speaker*, then attributes the quote to the top candidate.

Preprocessing: Finding candidate characters

The first preprocessing step is to identify the candidate speakers by "chunking" names (such as *Mr. Holmes*) and nominals (*the clerk*). We handle names and nominals separately, and only consider those that occur outside quotations.

For names, we process each text with the Stanford NER tagger (Finkel, Grenager, and Manning 2005) and extract chunks of contiguous proper nouns (excluding "locations"). We wrote a method to find coreferents among proper names and link them together as the same entity. This method selectively removes certain words in long names, respecting titles and first/last name distinctions, in order to generate likely variants of the name. If the variant is found elsewhere in the text, it is assumed to be a coreferent, similar to Davis, Elson and Klavans (2003). For example, *Mr. Sherlock Holmes* is matched to instances of *Mr. Holmes*, *Sherlock Holmes*, *Sherlock* and *Holmes*.

A separate method chunks character nominals by using a regular expression that searches each line for a determiner, an optional modifier, and a head noun. We compiled lists of determiners and head nouns using a subset of the development corpus: Determiners included the normal *a* and *the*, as well as possessives (*her father*, *Isabella's husband*) and both ordinal and cardinal numbers (*two women*). For legal head nouns, we used selected subtrees of the English taxonomy offered by WordNet (Fellbaum 1998), including organisms, imaginary beings and spiritual beings.[2]

We do not chunk pronouns as character candidates, because we would like the system (and the annotators) to dereference them back to the names to which they refer. We discuss below that about 9% of quotes are attributed to pronouns, and these cases reduce to an anaphora resolution problem. For this reason, during preprocessing we assign a gender to as many names and nominals as possible. We do this first through gendered titles (*Mr.*), gendered head words (*nephew*) and first names as given in a gendered name dictionary (*Emma*). Then, each referent of a named entity is assumed to share a gender with its assigned coreferents (e.g., *Mr. Scrooge* informs *Scrooge*). In case two referents for the same entity are marked with opposing genders by this heuristic, the system takes a majority vote among all the referents with assigned genders.

Encoding, cleaning, and normalizing

Before we extract features for each candidate-quote data point, we encode the passage between the candidate and the quote according to a backoff model. Our purpose here is to increase the amount of data that subscribes to similar patterns by substituting generic words and phrases for specific ones. The steps include:

1. Replacing the quote and character mention in question (the *target quote* and *target character*), as well as other quotes and characters, with symbols.
2. Replacing verbs that indicate verbal expression or thought with a single symbol, <EXPRESS_VERB>. We compiled the list of expression verbs by taking certain WordNet subtrees, similar to the manner in which we compiled character head nouns. We selected the subtrees based on the development corpus; they include certain senses of *express*, *think*, *talk* and *interrupt*, among others. There are over 6,000 words on this list in all, including various capitalized and conjugated forms for each verb.
3. Removing extraneous information, in particular adjectives, adverbs, and adverbial phrases. We identified these by processing the passage with the MXPOST part-of-speech tagger (Ratnaparkhi 1996).
4. Removing paragraphs, sentences and clauses where no information pertaining to quoted speech attribution seems to occur (e.g., no quotes, pronouns or names appear).

[1] http://www.cs.columbia.edu/nlp/tools.cgi

[2] The WordNet "organism" hierarchy includes many words not typically used as nouns, such as *heavy* in reference to "an actor who plays villainous roles." We improved precision by inserting a filter based on a rule-based classifier which we trained on a subset of the development corpus. Features included the numbers of WordNet senses for the word as an adjective, a noun and a verb, as well as the position of the organism sense among all the noun senses.

Syntactic category	Definition	Rate	Prediction	Accuracy
Backoff	n/a	.19		
Added quote	<OTHER_QUOTE by PERSON_1> <TARGET_QUOTE>	.19	PERSON_1	.95
Apparent conversation Multiple quotes appear in sequence without attribution.	<OTHER_QUOTE by PERSON_1> <OTHER_QUOTE by PERSON_2> <TARGET_QUOTE>	.18	PERSON_1	.96
Quote-Said-Person trigram	<TARGET_QUOTE> <EXPRESS_VERB> <PERSON_1>	.17	PERSON_1	.99
Quote alone	Quote appears by itself in a paragraph but "Apparent conversation" does not apply.	.14		
Anaphora trigram	<TARGET_QUOTE> <PRONOUN> <EXPRESS_VERB>	.10		
Quote-Person-Said trigram	<TARGET_QUOTE> <PERSON_1> <EXPRESS_VERB>	.02	PERSON_1	.92

Table 2: The most prevalent syntactic categories found in the development corpus.

Dialogue chains

One crucial aspect of the quote attribution task is that an author will often produce a sequence of quotes by the same speaker, but only attribute the first quote (at the head of the *dialogue chain*) explicitly. The effect of this discourse feature is that instances of quoted speech lack conditional independence. That is, the correct classification of one quote often depends on the correct classification of at least one previous quote. We read the text in a linear fashion and attribute quotes as we go, maintaining a discourse model that includes the currently speaking characters. For example:

"Bah!" said Scrooge, "Humbug!"

The added "Humbug" is implied to be spoken by the same speaker as "Bah." In general, the reader assumes that an "added" quote is spoken by the previous speaker, and that if several unattributed quotes appear in sequential paragraphs, they are two "intertwined" chains with alternating speakers. This model of reading is not tied to these authors or to this genre, but is rather a common stylistic approach to reporting conversational dialogue.

We model this dependence in both development and testing. In training statistical learners, we incorporate the annotations of speakers into the input features for subsequent quotes. In other words, the learner knows for each quote who spoke the previous quote. As the system processes a new text online, it solve quotes cumulatively from the front of the text to the back, just as a human reader would. During the backoff encoding, we include the identity of each previous speaker in its respective <OTHER_QUOTE> tag (see Table 2). This technique has the potential to propagate an error in attributing the "head" quote of a chain to the entire chain; in the present study we evaluate each quote under the ideal condition where previous quotes are correctly identified. We are currently investigating techniques for repairing discourse-level attribution errors.

Syntactic categories

Our approach is to classify the quotes and their passages in order to leverage two aspects of the semantics of quoted speech: dialogue chains and the frequent use of expression verbs. A pattern matching algorithm assigns to each quote one of five syntactic categories (see Table 2):

- **Added quote.** Intended for links in dialogue chains, this category covers quotes that immediately follow other quotes without paragraph breaks (e.g., "Humbug!").
- **Quote alone.** A quote appears by itself in a paragraph, without an attribution. In a subcategory, **apparent conversation**, two previous paragraphs begin with quotes that are either also alone or followed by sentences without quoted speech. This case is designed to correspond to alternating dialogue chains.
- **Character trigram.** This is a sequence of three adjacent tokens: a character mention, an expression verb and a span of quoted speech. There are six subcategories, one for each permutation (e.g., *"Bah!" said Scrooge* would be in the **Quote-Said-Person** subcategory, where "Said" refers to any expression verb and "Person" refers to a character mention).
- **Anaphora trigram.** There are six subcategories here that correspond to the six character trigrams, except that a pronoun takes the place of a character mention. Each subcategory is coded with the gender implied by the pronoun (male, female or plural speaker).
- **Backoff.** This catch-all category covers all quotes that are not covered by another category.

Two of these categories automatically imply a speaker for the quote. In *Added quote*, the speaker is the same as the one who spoke the preceding quote, and in character trigram categories, the mentioned character is the speaker. We shall see that these implied answers are highly accurate and sometimes obviate the need for machine learning for their respective categories. We divide the remaining cases into three data sets for learning: *No apparent pattern*, *Quote alone*, and any of the *Anaphora* trigrams. During online quote attribution, the syntactic classifier acts as a "router" that directs each quote to either a rapidly implied answer or one of the three models compiled by learners.

These categories are general enough to serve many genres of text that involve quoted speech. While we implemented the classifier using our development corpus, they are not designed for these authors or for 19th century texts in particular. However, the backoff category is used least often in conventional Western literary discourse that uses dialogue chains in the fashion described earlier. Some genres, such as epic poetry or 20th century modernism, vary in form to

"A merry Christmas, uncle! God save you!" cried **a cheerful voice**. It was **the voice** of **Scrooge's nephew**, who came upon him so quickly that this was the first intimation he had of his approach. "Bah!" said **Scrooge**, "Humbug!" He had so heated himself with rapid walking in the fog and frost, this nephew of Scrooge's, that he was all in a glow; his face was ruddy and handsome; his eyes sparkled, and his breath smoked again. *"Christmas a humbug, uncle!"* said **Scrooge's nephew.** "You don't mean that, I am sure?"	"And," said **Madame Bovary**, taking her watch from her belt, "take this; you can pay yourself out of it." But **the tradesman** cried out that she was wrong; they knew one another; did he doubt her? What childishness! She insisted, however, on his taking at least the chain, and **Lheureux** had already put it in his pocket and was going, when she called him back. *"You will leave everything at your place. As to the cloak"* – she seemed to be reflecting – "do not bring it either; you can give me the maker's address, and tell him to have it ready for me."
"Well, I do, too– LIVE ones. But I mean dead ones, to swing round your head with a string." "No, I don't care for rats much, anyway. What I like is chewing-gum." "Oh, I should say so! I wish I had some now." *"Do you? I've got some. I'll let you chew it awhile, but you must give it back to me."*	He beckoned coaxingly to **the Pomeranian**, and when **the dog** came up to him he shook his finger at it. **The Pomeranian** growled: **Gurov** shook his finger at it again. **The lady** looked at him and at once dropped her eyes. "He doesn't bite," she said, and blushed. *"May I give him a bone?"* he asked; and when she nodded he asked courteously, "Have you been long in Yalta?"

Table 3: Four samples of output that show the extracted character names and nominals (in bold).

the point where our system would place most quotes in the backoff category; however, for large volumes of literature (especially that which is available in electronic form), the categories apply.

Feature extraction and learning

To build these three predictive models, we extract a feature vector $\vec{f}$ for each candidate-quote pair. The features include:

- The distance (in words) between the candidate and quote
- The presence and type of punctuation between the candidate and quote (including paragraph breaks)
- Among the characters found near the quote, the ordinal position of the candidate outward from the quote. (In the anaphora cases, only gender-matching characters are counted)
- The proportion of the recent quotes that were spoken by the candidate
- Number of names, quotes, and words in each paragraph
- Number of appearances of the candidate
- For each word near the candidate and the quote, whether the word is an expression verb, a punctuation mark, or another person
- Various features of the quote itself, including the length, the position in the paragraph, the presence or absence of characters named within

Because this problem is one of choosing between candidates, we explore several ways of comparing each candidate's feature vector to those of its competitors within a set for a single quote. Specifically, we calculate the average value for each feature across the set and assemble a vector $\vec{f}_{mean}$. We then replace the absolute values for each candidate ($\vec{f}$) with the *relative distance* in value for each feature from the set norm, $\vec{f} - \vec{f}_{mean}$. We similarly experiment with sending $\vec{f} - \vec{f}_{median}$, $\vec{f} - \vec{f}_{product}$, $\vec{f} - \vec{f}_{max}$ and $\vec{f} - \vec{f}_{min}$ to the learners.

We applied three learners to the data. Each creates a model for predicting *speaker* or *non-speaker* given any candidate-quote feature vector. Namely, they are J48, JRip and a two-class logistic regression model with a ridge estimator, all as available in the WEKA Toolkit (Hall et al. 2009). Because these give binary labels and probability scores for each candidate separately, the final step is to **reconcile** these results into a single decision for each quote. We try four alternate methods:

- In the **label** method, we simply scan all candidates for one that has been classified *speaker*. If more than one candidate is classified *speaker*, the attribution remains ambiguous. If no speaker is found, the quote is determined to be non-dialogue. Overattributions (where a speaker is given to non-dialogue), underattributions (where no speaker is identified for dialogue) and misattributions (where the wrong speaker is identified) all count as errors.

- The **single probability** method discards the labels and simply uses the probability, supplied by each classifier, that each candidate belongs in the *speaker* class. When these probabilities are ranked, the candidate with the highest probability is taken as the speaker – unless the probability falls below a certain threshold, in which case we conclude the quote is non-dialogue (no speaker). We vary the threshold t as a parameter.

- The **hybrid** method works the same as the "label" method, except in case more than one candidate is labeled as *speaker*, the algorithm backs off to the single-probability method to find the best choice.

- The **combined probability** method works the same as the single probability method, except the probability of each candidate being *speaker* is derived by combining two or three of the probabilities given by the classifiers. We ran all permutations of classifiers and combined their results in four ways: mean, median, product and maximum, as suggested by Kittler et al. (1998).

Syntactic category	Rate	Solver	Feature vector	Reconciliation method	% correct
Quote-Said-Person	.22	**Category prediction**			**.99**
		Logistic+J48	$\vec{f} - \vec{f}_{min}$	Maximum ($t = .02$)	.96
Added quote	.19	**Category prediction**			**.97**
		J48	$\vec{f}$	Hybrid	.97
Backoff	.18	**Logistic+J48+JRip**	$\vec{f}$	Mean ($t = .08$)	**.64**
Quote alone	.16	**Logistic+J48+JRip**	$\vec{f} - \vec{f}_{mean}$	Mean ($t = .03$)	**.63**
Apparent conversation	.12	**JRip**	$\vec{f} - \vec{f}_{min}$	Hybrid	**.93**
		Category prediction			.91
Anaphora trigram	.09	**Logistic**	$\vec{f} - \vec{f}_{mean}$	Mean ($t = .01$)	**.63**
Quote-Person-Said	.04	**JRip**	$\vec{f}$	Hybrid	**.97**
		Category prediction			.93
Overall	1.0	In bold above			**.83**
Baseline	1.0	Most recent			.45
Baseline	1.0	Closest			.52

Table 4: Performance of both category predictions and learning tools on the test set for each syntactic category.

Results and discussion

Our results fall into three areas: the performance of our name and nominal chunker by itself; the impact of the categories and the answers they imply; and the combined accuracy on the test set, including the statistical learning.

Successful name chunking

We built a name and nominal chunker for this project because we found that available tools were not well-suited for 19th century fiction. We have found our method to have a very high recall (that is, it finds most names that become speaking characters). We quantify this by noting that only about 3% of votes cast in our gold-standard collection indicated that the correct answer was not properly extracted from the source text. The remaining 97% of votes indicated that if the speaker was mentioned in the passage, it was chunked and made available. Table 3 shows excerpts from Dickens, Flaubert, Chekhov and Twain (clockwise from top left), including names and nominals in bold that our system extracted as candidates. Again clockwise from top left, the syntactic categories for these passages are: *Quote-Said-Person*, *No apparent pattern*, *Quote-He-Said* (that is, *Anaphora*), and *Apparent conversation.*

Attribution results on testing corpus

Table 4 shows the performance of both the category predictions and the machine learning over the test set, with the latter using 10-fold cross-validation. Only the top-performing classifier permutation is shown for each category. For example, a combination of logistic regression, J48 and JRip, whose input features were absolute (rather than relative) and whose output probabilities were averaged before they were ranked, was trained and tested on all data in the backoff class and correctly identified the speaker (or lack of speaker) with 64% accuracy. Parameter tuning was done independently for each category. We achieved particularly high learning results in the categories where the speaker is determined by the category alone (such as *Added quote*); the decision tree learners are effectively deriving rules similar to those that we coded manually.

The *Rate* column in Table 4 shows the prevalence of each syntactic category in the testing corpus; these proportions differ only slightly from those in the development corpus (Table 2). When we sum the accuracy scores and weigh each according to their rates in the test set, we find an overall accuracy of .83. To ensure that we are not optimizing our classifier parameters for the test set, we separated out the parameter tuning process by having the test set adopt the classifier permutations that performed the best on the development set (one for each syntactic category). The overall accuracy over the test set with these learners was .80, suggesting that the classifier parameters are not overfitting the data. For purpose of comparison, a baseline that attributes a quote to the most recently seen character gives the correct speaker only in only .45 of cases. A smarter baseline that takes the closest occurring character, whether it appears before or after the quote, has an accuracy of only .52. Our results clearly show a significant improvement.

We can also view the data from another angle, that is, use quoted-speech attribution as a method for literary analysis. We mentioned earlier that Conan Doyle and Austen write a high proportion of quotes, while Flaubert and Twain write few quotes by comparison. We can also assess the complexity of each text by observing its style of quoting dialogue. Dickens, for example, writes almost half his quotes (47%) in the Quote-Said-Person category; the three "complex" categories only make up only 14% of his quotes (compared to 42% for the entire corpus). Conan Doyle's quotes are the second simplest. Flaubert, on the other hand, takes more of a concerted effort to untangle, with 62% of his quotes in the complex categories. At the same time, Flaubert has the lowest quote density of any work in the corpus, and the second-lowest share of back-and-forth conversations. (He was an early writer of italicized, "indirect" thought.) This is likely a reflection of his subject matter– a more disconnected, self-centered community. For Conan Doyle, though, clear and voluminous communication is essential to a proper mystery story so that the reader feels engaged and the detective may credibly solve the case. A full 29% of his quotes are in extended, play-like conversations.

Conclusion and Future Work

In this paper, we examined an important piece of the bridge between machine learning and literary analysis: how to automate the process of reading a text closely enough to disambiguate who is speaking or thinking each quotation. Our results exceeded the "nearest character" baseline, achieving 83% accuracy without knowing in advance who the candidate characters are in any given text. While modernist, experimental and verse texts are nonstandard, a wide array of literature can be processed this way. In the future, we plan to build on these results and move in the direction of social network extraction. In order to get a more complete picture of these and other texts, we also plan to investigate methods for extracting segments of indirect (unquoted) speech and their speakers.

Acknowledgments

We thank Fadi Biadsy, Nicholas Dames and Kapil Thadani for their comments on an earlier version of this paper. This material is based on research supported in part by the U.S. National Science Foundation (NSF) under IIS-0935360. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

References

Balahur, A.; Steinberger, R.; van der Goot, E.; Pouliquen, B.; and Kabadjov, M. 2009. Opinion mining on newspaper quotations. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology*.

Banfield, A. 1982. *Unspeakable sentences: narration and representation in the language of fiction*. Routledge.

Bird, C.; Gourley, A.; Devanbu, P.; Gertz, M.; and Swaminathan, A. 2006. Mining email social networks. In *Proceedings of the Third International Workshop on Mining Software Repositories (MSR 06)*.

Davis, P. T.; Elson, D. K.; and Klavans, J. L. 2003. Methods for precise named entity matching in digital collections. In *Proceedings of the Third ACM/IEEE Joint Conference on Digital Libraries (JCDL '03)*.

Fellbaum, C. 1998. *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press.

Finkel, J. R.; Grenager, T.; and Manning, C. D. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, 363–370.

Glass, K., and Bangay, S. 2007. A naive, salience-based method for speaker identification in fiction books. In *Proceedings of the 18th Annual Symposium of the Pattern Recognition Association of South Africa (PRASA '07)*, 1–6.

Grishman, R.; Westbrook, D.; and Meyers, A. 2005. Nyu's english ace 2005 system description. In *ACE 05 Evaluation Workshop*.

Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; and Witten, I. H. 2009. The weka data mining software: an update. *SIGKDD Explorations* 11(1).

Kittler, J.; Hatef, M.; Duin, R. P.; and Matas, J. 1998. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20(3):226–239.

Mamede, N., and Chaleira, P. 2004. Character identification in children stories. In *EsTAL 2004 - Advances in Natural Language Processing, LNCS*, 82–90. Berlin Heidelberg: Springer.

Pouliquen, B.; Steinberger, R.; and Best, C. 2007. Automatic detection of quotations in multilingual news. In *Proceedings of Recent Advances in Natural Language Processing 2007*.

Ratnaparkhi, A. 1996. A maximum entropy part-of-speech tagger. In *In Proceedings of the Empirical Methods in Natural Language Processing Conference*. University of Pennsylvania.

Redeker, G., and Egg, M. 2006. Says who? on the treatment of speech attributions in discourse structure. In *Proceedings of Constraints in Discourse*.

Salesin, D. K. T. S. D. 1996. Comic chat. In *Proceedings of SIGGRAPH '96*.

Sarmento, L., and Nunes, S. 2009. Automatic extraction of quotes and topics from news feeds. In *4th Doctoral Symposium on Informatics Engineering*.

Voloshinov, V. N. 1971. Reported speech. In Matejka, L., and Pomorska, K., eds., *Readings in Russian poetics: Formalist and structuralist views*. Cambridge: MIT Press. 149–175.

Wiebe, J. M. 1990. Identifying subjective characters in narrative. In *Proceedings of the 13th International Conference on Computational Linguistics (COLING-90)*, 401–408.

Zhang, J.; Black, A.; and Sproat, R. 2003. Identifying speakers in children's stories for speech synthesis. In *Proceedings of EUROSPEECH 2003*.