A Sequence Labelling Approach to Quote Attribution

Tim O'Keefe[†] **Silvia Pareti**[◇] **James R. Curran**[†] **Irena Koprinska**[†] **Matthew Honnibal**[‡]

[†]ə-lab, School of IT
University of Sydney
NSW 2006, Australia
{tokeefe,james,irena}@it.usyd.edu.au

[◇]School of Informatics
University of Edinburgh
United Kingdom
S.Pareti@sms.ed.ac.uk

[‡]Centre for Language Technology
Macquarie University
NSW 2109, Australia
matthew.honnibal@mq.edu.au

Abstract

Quote extraction and attribution is the task of automatically extracting quotes from text and attributing each quote to its correct speaker. The present state-of-the-art system uses gold standard information from previous decisions in its features, which, when removed, results in a large drop in performance. We treat the problem as a sequence labelling task, which allows us to incorporate sequence features without using gold standard information. We present results on two new corpora and an augmented version of a third, achieving a new state-of-the-art for systems using only realistic features.

1 Introduction

News stories are often driven by the quotes made by politicians, sports stars, musicians, and celebrities. When these stories exit the news cycle, the quotes they contain are often forgotten by both readers and journalists. A system that automatically extracts quotes and attributes those quotes to the correct speaker would enable readers and journalists to place news in the context of all comments made by a person on a given topic.

Though quote attribution may appear to be a straightforward task, the simple rule-based approaches proposed thus far have produced disappointing results. Going beyond these to machine learning approaches presents several problems that make quote attribution surprisingly difficult. The main challenge is that while a large portion of quotes can be attributed to a speaker based on simple rules, the remainder have few or no contextual clues as to who the correct speaker is. Additionally, many quote sequences, such as dialogues, rely on the reader understanding that there is an alternating sequence of speakers, which creates dependencies between attribution decisions made by a classifier.

Elson and McKeown (2010) is the only study that directly uses machine learning in quote attribution, treating the task as a classification task, where each quote is attributed independently of other quotes. To handle conversations and similar constructs they use gold standard information about speakers of previous quotes as features for their model. This is an unrealistic assumption, since gold standard information is not available in practice.

The primary contribution of this paper is that we reformulate quote attribution as a sequence labelling task. This allows us to use sequence features without having to use the unrealistic gold standard features that were used in Elson and McKeown (2010). We experiment with three sequence decoding models including greedy, Viterbi and a linear chain Conditional Random Field (CRF).

Furthermore we present results on two new corpora and an augmented version of a third. The two new corpora are from news articles from the Wall Street Journal and the Sydney Morning Herald respectively, while the third corpus is an extension to the classic literature corpus from Elson and McKeown (2010). Our results show that a quote attribution system using only realistic features is highly feasible for the news domain, with accuracies of 92.4% on the SMH corpus and 84.1% on the WSJ corpus.

2 Background

Early work into quote attribution by Zhang et al. (2003) focused on identifying when different characters were talking in children's stories, so that a speech synthesis system could read the quoted parts in different voices. While they were able to extract quotes with high precision and recall, their attribution accuracy was highly dependent on the document in question, ranging from 47.6% to 86.7%. Mamede and Chaleira (2004) conducted similar research on children's stories written in Portuguese. Their system proved to be very good at extracting quotes through simple rules, but when using a handcrafted decision tree to attribute those quotes to a speaker, they achieved an accuracy of only 65.7%.

In the news domain, both Pouliquen et al. (2007) and Sarmento and Nunes (2009) proposed rule-based systems that work over large volumes of text. Both systems aimed for high precision at the expense of low recall, as their data contained many redundant quotes. More recently, SAPIENS, a French-language quote extraction and attribution system, was developed by de La Clergerie et al. (2011). It conducts a full parse of the text, which allows it to use patterns to extract direct and indirect quotes, as well as the speaker of each quote. Their evaluation found that 19 out of 40 quotes (47.5%) had a correct span and author, while a further 19 had an incorrect author, and 4 had an incorrect span. In related work, Sagot et al. (2010) built a lexicon of French reported speech verbs, and conducted some analysis of different types of quotes.

Glass and Bangay (2007) approached the task with a three stage method. For each quote they first find the nearest speech verb, they then find the grammatical actor of that speech verb, and finally they select the appropriate speaker for that actor. To achieve each of these subtasks they built a model with several manually weighted features that good candidates should possess. For each subtask they then choose the candidate with the largest weighted sum of features. Their full approach yields an accuracy of 79.4% on a corpus of manually annotated fiction books.

Schneider et al. (2010) describe PICTOR, which is principally a quote visualisation tool. Their task was to find direct and indirect quotes, which they attribute to a text span representing the speaker. To do this they constructed a specialised grammar, which was built with reference to a small development corpus. With a permissive evaluation metric their grammar-based approach yielded 86% recall and 75% precision, however this dropped to 52% recall and 56% precision when measured in terms of completely correct quote-speaker pairs.

The work most similar to ours is the work by Elson and McKeown (2010). Their aim was to automatically identify both quotes and speakers, and then to attribute each quote to a speaker, in a corpus of classic literature that they compiled themselves. To identify potential speakers they used the Stanford NER tagger (Finkel et al., 2005) and a method outlined in Davis et al. (2003) that allowed them to find nominal character references. They then grouped name variants and pronominal mentions into a coreference chain.

To attribute a quote to a speaker they first classified the quotes into categories. Several of the categories have a speaker explicit in their structure, so they attribute quotes to those speakers with no further processing. For the remaining categories, they cast the attribution problem as a binary classification task, where each quote-speaker pair has a "speaker" or "not speaker" label predicted by the classifier. They then reconciled these independent decisions using various techniques to produce a single speaker prediction for each quote. For the simple category predictions they achieved 93-99% accuracy, while for the more complicated categories they achieved 63-64%, with an overall result of 83% accuracy. This compares favourably with their rule-based baseline, which achieved an accuracy of 52%.

While the results of Elson and McKeown (2010) appear encouraging, they are misleading for two reasons. First their corpus does not include quotes where all three annotators chose different speakers. While these quotes include some cases where the annotators chose coreferent spans, it also includes cases of legitimate disagreement about the speaker. An automated system would likely find these cases challenging. Second both their category predictions and machine learning predictions rely on gold standard information from previous quotes, which is not available in practice. In our study we address both these issues.

	Proportion (%)			Accuracy (%)		
	LIT	WSJ	SMH	LIT	WSJ	SMH
Quote-Said-Person	17.9	20.2	3.1	98.9	99.8	99.1
Quote-Person-Said	2.8	6.1	16.6	97.7	97.0	98.5
Other Trigram	0.1	2.3	0.3	66.7	56.2	54.5
Quote-Said-Pronoun	1.9	0.1	0.0	38.6	100.0	0.0
Quote-Pronoun-Said	5.9	8.8	13.5	36.5	92.2	93.9
Other Anaphors	0.1	0.1	0.2	0.0	100.0	62.5
Added*	24.6	28.3	23.9	89.7	76.3	97.5
Backoff	11.0	33.9	32.3	-	-	-
Alone	18.0	0.2	9.7	-	-	-
Conversation*	17.7	0.2	0.3	85.2	0.0	8.3
Total	100.0	100.0	100.0	60.5	57.2	55.8

Table 1: The proportion of quotes in each category and the accuracy of the speaker prediction based on the category. The two categories marked with an asterisk (*) depend on previous decisions.

3 Corpora

We evaluate our methods on two new corpora coming from the news domain, and an augmented version of an existing corpus, which covers classic literature. They are described below.

3.1 Columbia Quoted Speech Attribution Corpus (LIT)

The first corpus we use was originally created by Elson and McKeown (2010). It is a set of excerpts from 11 fictional 19th century works by six well-known authors, split into 18 documents. In total it contains 3,126 quotes annotated with their speakers.

Elson and McKeown used an automated system to find named entity spans and nominal mentions in the text, with the named entities being linked to form a coreference chain (they did not link nominal mentions). The corpus was built using Amazon's Mechanical Turk, with three annotations per quote. To ensure quality, all annotations from poorly performing annotators were removed, as were quotes where each annotator chose a different speaker. Though excluding some quotes ensures quality annotations, it causes gaps in the quote chains, which is a problem for sequence labelling. Furthermore, the cases where annotators disagreed are likely to be challenging, so removing them from the corpus could make results appear better than they would be in practice.

To rectify this, we conducted additional annotation of the quotes that were excluded by the original authors. Two postgraduates annotated 654 additional quotes, with a raw agreement of 79% over 48 double-annotated quotes. Our annotators reported seeing some errors in existing annotations, so we had one annotator check 400 existing annotations for correctness. This additional check found that 92.5% of the quotes were correctly annotated.

3.2 PDTB Attribution Corpus Extension (WSJ)

Our next corpus is an extension to the attribution annotations found in the Penn Discourse TreeBank (PDTB). The original PDTB contains several forms of discourse, including assertions, beliefs, facts, and eventualities. These can be attributed to named entities or to unnamed, pronominal, or implicit sources. Recent work by Pareti (2012) conducted further annotation of this corpus, including reconstructing attributions that were only partially annotated, and introducing additional information. From this corpus we use only direct quotes and the directly quoted portions of mixed quotes, giving us 4,923 quotes.

For the set of potential speakers we use the BBN pronoun coreference and entity type corpus (Weischedel and Brunstein, 2005), with automatically coreferred pronouns. We automatically matched BBN entities to PDTB extension speakers, and included the PDTB speaker where no matching BBN entity could be found. This means an automatic system has an opportunity to find the correct speaker for all quotes in the corpus.

3.3 Sydney Morning Herald Corpus (SMH)

We compiled the final corpus from a set of news documents taken from the Sydney Morning Herald website[1]. We randomly selected 965 documents published in 2009 that were not obituaries, opinion pages, advertisements or other non-news stories. To conduct the annotation we employed 11 non-expert annotators via the outsourcing site Freelancer[2], as well as five expert annotators from our research group. A total of 400 news stories were double-annotated, with at least 33 double-annotated stories per annotator. Raw agreement on the speaker of each quote was high at 98.3%. These documents had already been annotated with named entities as part of a separate research project (Hachey et al., 2012), which includes manually constructed coreference chains. The resulting corpus contains 965 documents, with 3,535 quotes.

3.4 Corpus Comparisons

In order to compare the corpora we categorise the quotes into the categories defined by Elson and McKeown (2010), as shown in Table 1. We assigned quotes to these categories by testing (after text preprocessing) whether the quote belonged to each category, in the order shown below:

1. *Trigram* – the quote appears consecutively with a mention of an entity, and a reported speech verb, in any order;
2. *Anaphors* – same as above, except that the mention is a pronoun;
3. *Added* – the quote is in the same paragraph as another quote that precedes it;
4. *Conversation* – the quote appears in a paragraph on its own, and the two paragraphs preceding the current paragraph each contain a single quote, with alternating speakers;
5. *Alone* – the quote is in a paragraph on its own;
6. *Miscellaneous* – the quote matches none of the preceding categories. This category is called "Backoff" in Elson and McKeown (2010).

[1] http://www.smh.com.au

[2] http://www.freelancer.com

Unsurprisingly, the two corpora from the news domain share similar proportions of quotes in each category. The main differences are that the SMH uses a larger number of pronouns compared to the WSJ, which tends to use explicit attribution more frequently. The SMH also has a significant proportion of quotes that appear alone in a paragraph, while the WSJ has almost none. Finally, when attributing a quote using a trigram pattern, the SMH mostly uses the Quote-Person-Said pattern, while the WSJ mostly uses the Quote-Said-Person pattern. These differences probably reflect the editorial guidelines of the two newspapers.

The differences between the news corpora and the literature corpus are more substantial. Most notably the LIT corpus has a much higher proportion of quotes that fall into the *Conversation* and *Alone* categories. This is unsurprising as both monologues and dialogues are common in fiction, but are rare in newswire. The two news corpora have more quotes in the *Trigram* and *Backoff* categories.

4 Quote Extraction

Quote extraction is the task of finding the spans that represent quotes within a document. There are three types of quotes that can appear:

1. *Direct quotes* appear entirely between quotation marks, and are used to indicate that the speaker said precisely what is written;
2. *Indirect quotes* do not appear between or contain quotation marks, and are used to get the speaker's point across without implying that the speaker used the exact words of the quote;
3. *Mixed quotes* are indirect quotes that contain a directly quoted portion.

In this work, we limit ourselves to detecting direct quotes and the direct portions of mixed quotes.

To extract quotes we use a regular expression that searches for text between quotation marks. We also deal with the special case of multi-paragraph quotes where one quotation mark opens the quote and every new paragraph that forms part of the quote, with a final quotation mark only at the very end of the quote. This straightforward approach yields over 99% accuracy on all three corpora.

5 Quote Attribution

Given a document with a set of quotes and a set of entities, quote attribution is the task of finding the entity that represents the speaker of each quote, based on the context provided by the document. Identifying the correct entity can involve choosing either an entire coreference chain representing an entity, or identifying a specific span of text that represents the entity.

In practice, most applications only need to know which coreference chain represents the speaker, not which particular span in the text. Despite this, the best evidence about which chain is the speaker is found in the context of the individual text spans, and most existing systems aim to get the particular entity span correct. This presents a problem for evaluation, as an incorrect entity span may be identified, but it might still be part of the correct coreference chain. We chose to count attributions as correct if they attributed the quote to the correct coreference chain for both the LIT and SMH corpora, while for the WSJ corpus, where the full coreference chains do not exist, we evaluated an attribution as correct if it was to the correct entity span in the text.

5.1 Rule-based Baseline

To establish the effectiveness of our method we built a rule-based baseline system. For each quote it proceeds with the following steps:

1. Search backwards in the text from the end of the sentence the quote appears in for a reported speech verb
2. If the verb is found return the entity mention nearest the verb (ignoring mentions in quotes), in the current sentence or any sentence preceding it
3. If not, return the mention of an entity nearest the end of the quote (ignoring mentions in quotes), in the current sentence or any sentence preceding it

This forms a reasonable baseline as it is able to pick up the quotes that fall into the more simple categories, such as the *Trigram* category and the *Added* category. It is also able to make a guess at the more complicated categories, without using gold standard information as the category predictions do.

6 Experimental Setup

We use two classifiers: a logistic regression implementation available in LIBLINEAR (Fan et al., 2008), and a Conditional Random Field (CRF) from CRFSuite (Okazaki, 2007). Both packages use maximum likelihood estimation with L2 regularisation. We experimented with several values for the coefficient on a development set, but found that it had little impact, so stuck with the default value. All of our machine learning experiments use the same text encoding, which is explained below, and all use the category predictions when they are available.

6.1 Text Encoding

We encode our text similarly to Elson and McKeown (2010). The major steps are:

1. Replace all quotes and speakers with special symbols;
2. Replace all reported speech verbs with a symbol. Elson and McKeown (2010) provided us with their list of reported speech verbs;
3. Part-of-Speech (POS) tag the text and remove adjectives, adverbs, and other parts of speech that do not contribute useful information. We used the POS tagger from Curran and Clark (2003);
4. Remove any paragraphs or sentences where no quotes, pronouns or names occur.

All features that will be discussed are calculated with respect to this encoding (e.g. word distance would be the number of words in the encoded text, rather than the number of words in the original text).

6.2 Features

In our experiments we use the feature set from Elson and McKeown (2010). The features for a particular pair of target quote (q) and target speaker (s) are summarised below.

Distance features including number of words between q and s, number of paragraphs between q and s, number of quotes between q and s, and number of entity mentions between q and s

Corpus	Sequence Features		
	Gold	Pred	None
LIT	74.7	49.0	49.6
WSJ	87.3	74.1	82.9
SMH	95.0	85.6	92.4

Table 2: Accuracy results comparing the E&M approach with gold standard, predicted or no sequence features.

Paragraph features derived from the 10 paragraphs preceding the quote (including the paragraph the quote is in), includes number of mentions of s, number of mentions of other speakers, number of words in each paragraph, and number of quotes in each paragraph

Nearby features relating to the two tokens either side of q and s, includes binary features for each position indicating whether the position is punctuation, s, q, a different speaker, a different quote, or a reported speech verb

Quote features about q itself, including whether s is mentioned within it, whether other speakers are mentioned within it, how far the quote is from the start of its paragraph and the length in words of q

Sequence features that depend on the speakers chosen for the previous quotes, includes number of quotes in the 10 paragraphs preceding and including the paragraph where q appears that were attributed to s, and the number that were attributed to other speakers

6.3 Elson and McKeown Reimplementation

As part of our study we reproduce the core results of Elson and McKeown (2010) (E&M), as we believe it is a state-of-the-art system. This allows us to determine the effectiveness of our approach when compared to a state-of-the-art approach, and it also allows us to determine how well the E&M approach performs on other corpora. In this section we will briefly summarise the key elements needed to reproduce their work.

The E&M approach makes a binary classification between "speaker" and 'not speaker" for up to 15 candidate speakers for each quote. They then reconcile these 15 classifications into one speaker prediction for the quote. While E&M experimented with several different reconciliation methods, we simply chose the speaker with the highest probability attached to its "speaker" label.

We conducted an experiment using our implementation of the E&M method on the original, unaugmented E&M corpus, to see how our result compared with E&M 's 83%. On our test set we achieved 78.2%, however this rose to 82.3% when performing 10-fold cross validation across the whole corpus. Though this is a large difference, it is not necessarily that surprising, as our test set contains documents by authors which are unseen, whereas both the original E&M test set and all the cross validation test sets contain documents by authors that the learner has seen before.

In their work, E&M make a simplifying assumption that all previous attribution decisions were correct. Due to this, their sequence features use gold standard labels from previous quotes, which makes their results unrealistic. In Table 2 we show the effect of replacing the gold standard sequence features with features based on the predicted labels, or with no sequence features at all. All three corpora show a significant drop in accuracy, with the LIT corpus in particular suffering a drop of more than 25%. This motivates our study into including sequence information without using gold standard labels.

7 Class Models

We consider two class models for our experiments, which are described in detail below. The binary model is able to take advantage of more data but has less competition between decisions, while the n-way model has more competition with less data. Both models are used with all the decoding methods, with the exception that the binary model is unsuitable for the CRF experiments.

7.1 Binary

When working with n previous speakers, a binary class model works by predicting n independent "speaker" versus "not speaker" labels, one for each quote-speaker pair. As the classifications are independent the n decisions need to be reconciled, as more than one speaker might be predicted. We reconcile the n decisions by attributing the quote to the

speaker with the highest "speaker" probability. Using a binary class with reconciliation in a greedy decoding model is equivalent to the method in Elson and McKeown (2010), except that the gold standard sequence features are replaced with predicted sequence features.

7.2 n-way

A key advantage of the binary class model is that when predicting "speaker" versus "not speaker" the classifier only needs to predict one probability, and thus can take into account the evidence of all other quote-speaker pairs. The drawback to the binary model is that the probabilities assigned to the candidate speakers do not need to directly compete against each other. In other words when assigning a binary probability to a candidate speaker, the classifier does not take into account how good the other candidate speakers are.

To rectify these issues we experiment with a single classification for each quote, where the classifier directly decides between up to n candidate speakers per quote. As speaker-specific evidence is far too sparse, we encode the speakers with their ordinal position backwards from the quote. In other words, the candidate speaker immediately preceding the quote would be labelled "speaker1", the speaker preceding it would be "speaker2" and so on. The classifier then directly predicts these labels. This representation means that candidate speakers need to directly compete for probability mass, although it has the drawback that the evidence for the higher-numbered speakers is quite sparse.

The features we use for this representation are similar to the features used in the E&M binary model. The key difference is that where there were individual features that were calculated with respect to the speaker, there are now n features, one for each of the speaker candidates. This allows the model to account for the strength of other candidates when assigning a speaker label.

8 Sequence Decoding

We noted in the previous section that the E&M results are based on the unrealistic assumption that all previous quotes were attributed correctly. In this section we outline three sequence decoding approaches that remove this unrealistic assumption, without removing all of the transition information that it provides. We believe the transition information is important as many quotes have no explicit attribution in the text, and instead rely on the reader understanding something about the sequence of speakers.

For these experiments we regard the set of speaker attributions in a document as the sequence that we want to decode. Each individual state therefore represents a sequence of w previous attribution decisions, and a decision for the current quote. Obtaining a probability for this state can be done in one of two ways. Either the transition probabilities from state to state can be learned explicitly, or the w previous attribution decisions can be used to build the sequence features for the current state, which implicitly encodes the transition probabilities.

8.1 Greedy Decoding

In sequence decoding the greedy algorithm calculates the probability of each label at a decision point based on the predictions it has already made for previous decisions. More concretely this means we apply a standard classifier at each step, with the sequence features being calculated from the predictions made in previous steps. Greedy decoding is efficient in that it only considers one possible history at each decision point, but it is consequently unable to make trade-offs between good previous choices and good current choices, which means that in general it will not return the optimum sequence of labels. As greedy decoding is an efficient algorithm we do not restrict w, the number of previous decisions, beyond the 10 paragraph restriction that is already in place.

8.2 Viterbi Decoding

Viterbi decoding finds the most probable path through a sequence of decisions. It does this by determining the probabilities of each of the labels at the current decision point, with each of the possible histories of decisions within a given window w. These probabilities can be multiplied together with the previous decisions to retrieve a joint probability for the entire sequence. The final decision for each quote is then just the speaker which is predicted by the sequence with the largest joint probability.

Although they do not come with probabilities, we chose to include the category predictions in our Viterbi model. As we already know that they are accurate indicators of the speaker we assign them a probability of 100%, which effectively forces the Viterbi decoder to choose the category predictions when they are available. It is worth noting that quotes are only assigned to the *Conversation* category if the two prior quotes had alternating speakers. As such, during the Viterbi decoding the categorisation of the quote actually needs to be recalculated with regard to the two previous attribution decisions. By forcing the Viterbi decoder to choose category predictions when they are available, we get the advantage that quote sequences with no intervening text may be forced into the *Conversation* category, which is typically under-represented otherwise.

Both the sequences using the binary class and the n-way class can be decoded using the Viterbi algorithm, so we experiment with both class models. We also experiment with varying window sizes (w), in order to gain insight into how many previous decisions impact the current decision. Though the Viterbi algorithm is able to find the best sequence of probabilities without the need for an exhaustive search, it can still take an impractical amount of time to run. As such we ignore all but the 10 most promising sequences at each decision point.

8.3 Conditional Random Field (CRF) Decoding

The key drawback with the logistic regression experiments described thus far is that the sequence features are trained with gold standard information. This means that during the training phase the sequence features have perfect information about previous speakers and are thus unrealistically good predictors of the final outcome. When the resulting model is used with the less accurate predicted sequence features, it is overconfident about the information those features provide.

We account for this by using a first-order linear chain CRF model, which learns the probabilities of progressing from speaker to speaker more directly. During training the CRF is able to learn the association between features and labels, as well as the chance of transitioning from one label to the next. It also has the advantage of avoiding the label bias problem that would be present in the equivalent Hidden Markov Model (Lafferty et al., 2001).

Though the n-way class model can be used directly in a CRF, the binary class model is more challenging. The main problem is that the "speaker" versus "not speaker" output of the binary classifier does not directly form a meaningful sequence that the CRF can learn over. If the reconciliation step is included it effectively adds an extra layer to the linear chain, making learning more difficult. Due to these difficulties we only use the n-way class model in our CRF experiments.

9 Results

The main result of our experiments with the E&M method is the large drop in accuracy that occurs when the gold standard sequence features are removed, which can be seen in Table 3. When using the binary class model this results in a drop of 25.1% for the LIT corpus, while for the WSJ and SMH corpora the drop is less substantial at 4.4% and 2.6%, respectively. For the LIT corpus the drop is so severe that it actually performs worse than the simple rule-based system. Even more surprisingly, when the predictions from previous decisions are used with a simple greedy decoder, the accuracy drops even further for all three corpora. This indicates that the classifier is putting too much weight on the gold standard sequence features during training, and is misled into making poor decisions when the predicted features are used during test time.

Table 4 shows the results for the n-way class model. Compared to the binary model, the n-way class model generally produced lower results, although the results were more stable to changes in parameters and decoders. The only corpus that produced better results with the n-way class model was the WSJ corpus, which does not have full entity coreference information. This indicates that the n-way model may be helpful when there is more variety in the choice of entities.

The final results we would like to discuss here are the CRF results. On all three corpora the CRF results are underwhelming. The major issue that we can see when applying a CRF model to this task is that the sequences that it needs to learn over are entire documents. This means that for the LIT corpus the training set consisted of only 12 sequences, while

Corpus	E&M	Rule	No seq.	Greedy	Viterbi		
					$w = 1$	$w = 2$	$w = 5$
LIT	*74.7*	**53.3**	49.6	49.0	46.0	49.8	45.9
WSJ	*87.3*	77.9	82.9	74.1	82.3	**83.1**	**83.1**
SMH	*95.0*	91.2	**92.4**	85.6	91.7	90.5	84.1

Table 3: Accuracy on test set with the binary class model. Italicised results indicate gold standard information is used. Bold results show the best realistic result for each corpus.

Corpus	Gold seq.	Rule	No seq.	Greedy	Viterbi			CRF
					$w = 1$	$w = 2$	$w = 5$	
LIT	*68.6*	**53.3**	47.1	46.7	42.5	46.5	44.4	48.6
WSJ	*88.9*	77.9	83.6	77.0	**84.1**	83.7	83.3	79.6
SMH	*94.4*	**91.2**	90.0	89.6	89.5	90.1	90.4	91.0

Table 4: Accuracy on test set with the n-way class model. Italicised results indicate gold standard information is used. Bold results show the best realistic result for each corpus.

the test set consisted of 6 sequences. With so few sequences it is unsurprising that the CRF model did not perform well. The limited range of the first order linear chain model could also have played a part in the poor performance of the CRF models. However, moving to a higher-order model is problematic as the number of transition probabilities that need to be calculated increases exponentially with the order of the model.

10 Conclusion

In this paper, we present the first large-scale evaluation of a quote attribution system on newswire from the 1989 Wall Street Journal (WSJ) and the 2009 Sydney Morning Herald (SMH), as well as comparing against previous work (Elson and McKeown, 2010) on 19th-century literature.

We show that when Elson and McKeown's unrealistic use of gold-standard history information is removed, accuracy on all three corpora drops substantially. We demonstrate that by treating quote attribution as a sequence labelling task, we can achieve results that are very close to their results on newswire, though not for literature.

In future work, we intend to further explore the sequence features that have a large impact on accuracy, and to find similar features or proxies for the sequence features that would be beneficial. We will also explore other approaches to representing quote attribution with a CRF. For the task more broadly, it would be beneficial to compare methods of finding indirect and mixed quotes, and to evaluate how well quote attribution performs on those quotes as opposed to just direct quotes.

Our newswire results, 92.4% for the SMH and 84.1% for the WSJ corpus, demonstrate it is possible to develop an accurate and practical quote extraction system. On the LIT corpus our best result was from the simple rule-based system, which yielded 53.3%. It is clear that literature poses an ongoing research challenge.

Acknowledgements

We would like to thank David Elson for helping us to reimplement his method and Bonnie Webber for her feedback and assistance. O'Keefe has been supported by a University of Sydney Merit scholarship and a Capital Markets CRC top-up scholarship; Pareti has been supported by a Scottish Informatics and Computer Science Alliance (SICSA) studentship. This work has been supported by ARC Discovery grant DP1097291 and the Capital Markets CRC Computable News project.

References

James R. Curran and Stephen Clark. 2003. Investigating GIS and smoothing for maximum entropy taggers. In *Proceedings of the tenth conference on*

European chapter of the Association for Computational Linguistics, pages 91–98.

Peter T. Davis, David K. Elson, and Judith L. Klavans. 2003. Methods for precise named entity matching in digital collections. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital libraries*, pages 125–127.

Eric de La Clergerie, Benoit Sagot, Rosa Stern, Pascal Denis, Gaelle Recource, and Victor Mignot. 2011. Extracting and visualizing quotations from news wires. *Human Language Technology. Challenges for Computer Science and Linguistics*, pages 522–532.

David. K Elson and Kathleen. R McKeown. 2010. Automatic attribution of quoted speech in literary narrative. In *Proceedings of AAAI*, pages 1013–1019.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370.

Kevin Glass and Shaun Bangay. 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 (PRASA07)*, pages 1–6.

Ben Hachey, Will Radford, Joel Nothman, Matthew Honnibal, and James R. Curran. 2012. Evaluating entity linking with Wikipedia. *Artificial Intelligence*. (in press).

John Lafferty, Andrew McCallum, and Fernando C.N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *International Conference on Machine Learning*, pages 282–289.

Nuno Mamede and Pedro Chaleira. 2004. Character identification in children stories. *Advances in Natural Language Processing*, pages 82–90.

Naoaki Okazaki. 2007. CRFsuite: a fast implementation of Conditional Random Fields (CRFs). URL `http://www.chokkan.org/software/crfsuite/`.

Silvia Pareti. 2012. A database of attribution relations. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 3213–3217.

Bruno Pouliquen, Ralf Steinberger, and Clive Best. 2007. Automatic detection of quotations in multilingual news. In *Proceedings of Recent Advances in Natural Language Processing*, pages 487–492.

Benoît Sagot, Laurence Danlos, and Rosa Stern. 2010. A lexicon of french quotation verbs for automatic quotation extraction. In *7th international conference on Language Resources and Evaluation - LREC 2010*.

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

Nathan Schneider, Rebecca Hwa, Philip Gianfortoni, Dipanjan Das, Michael Heilman, Alan W. Black, Frederik L. Crabbe, and Noah A. Smith. 2010. Visualizing topical quotations over time to understand news discourse. Technical Report CMU-LTI-01-013, Carnegie Mellon University.

Ralph Weischedel and Ada Brunstein. 2005. BBN pronoun coreference and entity type corpus. *Linguistic Data Consortium, Philadelphia*.

Jason Zhang, Alan Black, and Richard Sproat. 2003. Identifying speakers in children's stories for speech synthesis. In *Proceedings of EUROSPEECH*, pages 2041–2044.