A Joint Model for Quotation Attribution and Coreference Resolution

Mariana S. C. Almeida[*†] **Miguel B. Almeida**[*†] **André F. T. Martins**[*†]
[*]Priberam Labs, Alameda D. Afonso Henriques, 41, 2º, 1000-123 Lisboa, Portugal
[†]Instituto de Telecomunicações, Instituto Superior Técnico, 1049-001 Lisboa, Portugal
{mla,mba,atm}@priberam.pt

Abstract

We address the problem of automatically attributing quotations to speakers, which has great relevance in text mining and media monitoring applications. While current systems report high accuracies for this task, they either work at mention-level (getting credit for detecting uninformative mentions such as pronouns), or assume the coreferent mentions have been detected beforehand; the inaccuracies in this preprocessing step may lead to error propagation. In this paper, we introduce a joint model for entity-level quotation attribution and coreference resolution, exploiting correlations between the two tasks. We design an evaluation metric for attribution that captures all speakers' mentions. We present results showing that both tasks benefit from being treated jointly.

1 Introduction

Quotations are a crucial part of news stories, giving the perspectives of the participants in the narrated event, and making the news sound objective. The ability of extracting and organizing these quotations is highly relevant for text mining applications, as it may aid journalists in fact-checking, help users browse news threads, and reduce human intervention in media monitoring. This involves assigning the correct speaker to each quote—a problem called **quotation attribution** (§2).

There is significant literature devoted to this task, both for narrative genres (Mamede and Chaleira, 2004; Elson and McKeown, 2010) and newswire domains (Pouliquen et al., 2007; Sarmento et al., 2009; Schneider et al., 2010). While the earliest works focused on devising lexical and syntactic rules and hand-crafting grammars, there has been a recent shift toward machine learning approaches (Fernandes et al., 2011; O'Keefe et al., 2012; Pareti et al., 2013), with latest works reporting high accuracies for speaker identification in newswire (in the range 80–95% for direct and mixed quotes, according to O'Keefe et al. (2012)). Despite these encouraging results, quotation mining systems are not yet fully satisfactory, even when only direct quotes are considered. Part of the problem, as we next describe, has to do with inaccuracies in **coreference resolution** (§3).

The "easiest" instances of quotation attribution problems arise when the speaker and the quote are semantically connected, *e.g.*, through a reported speech verb like *said*. However, in newswire text, the subject of this verb is commonly a pronoun or another uninformative anaphoric mention. While the speaker thus determined may well be correct—being in most cases consistent with human annotation choices (Pareti, 2012)—from a practical perspective, it will be of little use without a coreference system that correctly resolves the anaphora. Since the current state of the art in coreference resolution is far from perfect, errors at this stage tend to propagate to the quote attribution system.

Consider the following examples for illustration (taken from the WSJ-1057 and WSJ-0089 documents in the Penn Treebank), where we have annotated with subscripts some of the mentions:

(a) Rivals carp at "the principle of [Pilson]$_{M_1}$," as [NBC's Arthur Watson]$_{M_2}$ once put it – *"[he]$_{M_3}$'s always expounding that rights are too high, then [he]$_{M_4}$'s going crazy."* But [the 49-year-old Mr. Pilson]$_{M_5}$ is hardly a man to ignore the numbers.

(b) [English novelist Dorothy L. Sayers]$_{M_1}$ described [ringing]$_{M_2}$ as a *"passion that finds its satisfaction in [mathematical completeness]$_{M_3}$ and [mechanical perfection]$_{M_4}$."* [Ringers]$_{M_5}$, [she]$_{M_6}$ added, are *"filled with the solemn intoxication that comes of intricate ritual faultlessly performed."*

In example (a), the pronoun coreference system used by O'Keefe et al. (2012) erroneously clusters together mentions M_2, M_3 and M_4 (instead of the correct clustering $\{M_1, M_3, M_4\}$). Since it is unlikely that the speaker is co-referent to a third-

Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pages 39–48,
Gothenburg, Sweden, April 26-30 2014.

person pronoun *he* inside the quote, a pipeline system would likely attribute (incorrectly) this quote to *Pilson*. In example (b), there are two quotes with the same speaker entity (as indicated by the cue *she added*). This gives evidence that M_1 and M_6 should be coreferent. A pipeline approach would not be able to exploit these correlations.

We argue that this type of mistakes, among others, can be prevented by a system that performs quote attribution and coreference resolution **jointly** (§4). Our joint model is inspired by recent work in coreference resolution that independently ranks the possible mention's antecedents, forming a latent coreference tree structure (Denis and Baldridge, 2008; Fernandes et al., 2012; Durrett et al., 2013; Durrett and Klein, 2013). We consider a generalization of these structures which we call a **quotation-coreference tree**. To effectively couple the two tasks, we need to go beyond simple arc-factored models and consider paths in the tree. We formulate the resulting problem as a logic program, which we tackle using a dual decomposition strategy (§5). We provide an empirical comparison between our method and baselines for each of the tasks and a pipeline system, defining suitable metrics for entity-level quotation attribution (§6).

2 Quotation Attribution

The task of quotation attribution can be formally defined as follows. Given a document containing a sequence of quotations, $\langle q_1, \ldots, q_L \rangle$, and a set of candidate speakers, $\{s_1, \ldots, s_M\}$, the goal is to a assign a speaker to every quote.

Previous work has handled direct and mixed quotations (Sarmento et al., 2009; O'Keefe et al., 2012), easily extractable with regular expressions for detecting quotation marks, as well as indirect quotations (Pareti et al., 2013), which are more involved and require syntactic or semantic patterns. In this work, we resort to direct and mixed quotations. Pareti (2012) defines quotation attributions in terms of their *content span* (the quotation text itself), their *cue* (a lexical anchor of the attribution relation, such as a reported speech verb), and the *source span* (the author of the quote). The same reference introduced the PARC dataset, which we use in our experiments (§6) and which is based on the annotation of a database of attribution relations from the Penn Discourse Treebank (Prasad et al., 2008). Several machine learning algorithms have been applied to this task, either framing the problem as classification (an independent decision for each quote), or sequence labeling (using greedy methods or linear-chain conditional random fields); see O'Keefe et al. (2012) for a comparison among these different methods.

In this paper, we distinguish between **mention-level** quotation attribution, in which the candidate speakers are individual mentions, and **entity-level** quotation attribution, in which they are entity clusters comprised of one or more mentions. With this distinction, we attempt to clarify how prior work has addressed this task, and design suitable baselines and evaluation metrics. For example, O'Keefe et al. (2012) applies a coreference resolver *before* quotation attribution, whereas de La Clergerie et al. (2011) does it *afterwards*, as a post-processing stage. An important issue when evaluating quotation attribution systems is to prevent them from getting credit for detecting uninformative speakers such as pronouns; we will get back to this topic in §6.2.

3 Coreference Resolution

In coreference resolution, we are given a set of mentions $M := \{m_1, \ldots, m_K\}$, and the goal is to cluster them into discourse entities, $E := \{e_1, \ldots, e_J\}$, where each $e_j \subseteq M$ and $e_j \neq \varnothing$. We follow Haghighi and Klein (2007) and distinguish between proper, nominal, and pronominal mentions. Each requires different types of information to be resolved. Thus, the task involves determining anaphoricity, resolving pronouns, and identifying semantic compatibility among mentions. To resolve these references, one typically exploits contextual and grammatical clues, as well as semantic information and world knowledge, to understand whether mentions refer to people, places, organizations, and so on. The importance of coreference resolution has led to it being the subject of recent CoNLL shared tasks (Pradhan et al., 2011; Pradhan et al., 2012).

There has been a variety of approaches for this problem. Early work used local discriminative classifiers, making independent decisions for each mention or pair of mentions (Soon et al., 2001; Ng and Cardie, 2002). Lee et al. (2011) proposed a competitive non-learned sieve-based method, which constructs clusters by aglomerating mentions in a greedy manner. Entity-centric models define scores for the entire entity clusters (Culotta et al., 2007; Haghighi and Klein, 2010;

Rahman and Ng, 2011) and seek the set of entities that optimize the sum of scores; this can also be promoted in a decentralized manner (Durrett et al., 2013). Pairwise models (Bengtson and Roth, 2008; Finkel et al., 2008; Versley et al., 2008), on the other hand, define scores for each pair of mentions to be coreferent, and define the clusters as the transitive closure of these pairwise relations. A disadvantage of these two methods is that they lead to intractable decoding problems, so approximate methods must be used. For comprehensive overviews, see Stoyanov et al. (2009), Ng (2010), Pradhan et al. (2011) and Pradhan et al. (2012).

Our joint approach (to be fully described in §4) draws inspiration from recent work that shifts from entity clusters to **coreference trees** (Fernandes et al., 2012; Durrett and Klein, 2013). These models define scores for each mention to link to its antecedent or to an artifical root symbol \$ (in which case it is not anaphoric). The computation of the best tree can be done exactly with spanning tree algorithms, or by independently choosing the best antecedent (or the root) for each mention, if only left-to-right arcs are allowed. The same idea underlies the antecedent ranking approach of Denis and Baldridge (2008). Once the coreference tree is computed, the set of entity clusters E is obtained by associating each entity set to a branch of the tree coming out from the root. This is illustrated in Figure 1 (left).

4 Joint Quotations and Coreferences

In this work, we propose that quotation attribution and coreference resolution are solved jointly by treating both mentions and quotations as nodes in a generalized structure called a **quotation-coreference tree** (Figure 1, right). The joint system's decoding process consists in creating such a tree, from which a clustering of the nodes can be immediatelly obtained. The clustering is interpreted as follows:

- All mention nodes in the cluster are coreferent, thus they describe one single entity (just like in a standard coreference tree).
- Quotation nodes that appear together with those mentions in a cluster will be assigned that entity as the speaker.

For example, in Figure 1 (right), the entity *Dorothy L. Sayers* (formed by mentions $\{M_1, M_6\}$) is assigned as the speaker of quotations Q_1 and Q_2. We forbid arcs between quotes and from a quote to a mention, effectively constraining the quotes to be *leaves* in the tree, with mentions as parents.[1] We force a tree with only left-to-right arcs, by choosing a total ordering of the nodes that places all the quotations in the rightmost positions (which implies that any arc connecting a mention to a quotation will point to the right). The quotation-coreference tree is obtained as the best spanning tree that maximizes a score function, to be described next.

4.1 Basic Model

Our basic model is a feature-based linear model which assigns a score to each candidate arc linking two mentions (*mention-mention arcs*), or linking a mention to a quote (*mention-quotation arcs*). Our basic system is called QUOTEBEFORECOREF for reasons we will detail in section 4.2.

4.1.1 Coreference features

For the mention-mention arcs, we use the same coreference features as the SURFACE model of the Berkeley Coreference Resolution System (Durrett and Klein, 2013), plus features for gender and number obtained through the dataset of Bergsma and Lin (2006). This is a very simple lexical-driven model which achieves state-of-the-art results. The features are shown in Table 1.

4.1.2 Quotation features

For the quote attribution features, we use features inspired by O'Keefe et al. (2012), shown in Table 2. The same set of features works for speakers that are individual mentions (in the model just described), and for speakers that are clusters of mentions (used in §6 for the baseline QUOTEAFTERCOREF). These features include various distances between the mention and the quote, the indication of the speaker being inside the quote span, and various contextual features.

4.2 Final Model

While the basic model just described puts quotations and mentions together, it is not more expressive than having separate models for the two tasks. In fact, if we just have scores for individual arcs, the two problems are *decoupled*: the optimal

[1] This is implemented by defining $-\infty$ scores for all the outgoing arcs in a quotation node, as well as incoming arcs originating from the root.

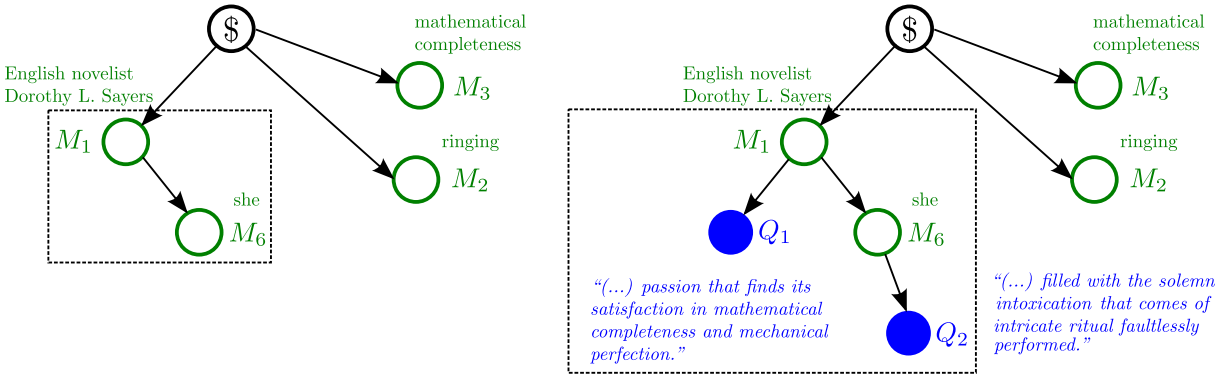

Figure 1: *Left:* A typical coreference tree for the text snippet in §1, example (b), with mentions M_1 and M_6 clustered together and M_2 and M_3 left as singletons. *Right:* A quotation-coreference tree for the same example. Mention nodes are depicted as green circles, and quotation nodes in shaded blue. The dashed rectangle represents a branch of the tree, containing the entity cluster associated with the speaker *Dorothy L. Sayers*, as well as the quotes she authored.

Features on the child mention
[ANAPHORIC (T/F)] + [CHILD HEAD WORD]
[ANAPHORIC (T/F)] + [CHILD FIRST WORD]
[ANAPHORIC (T/F)] + [CHILD LAST WORD]
[ANAPHORIC (T/F)] + [CHILD PRECEDING WORD]
[ANAPHORIC (T/F)] + [CHILD FOLLOWING WORD]
[ANAPHORIC (T/F)] + [CHILD LENGTH]
Features on the parent mention
[PARENT HEAD WORD]
[PARENT FIRST WORD]
[PARENT LAST WORD]
[PARENT PRECEDING WORD]
[PARENT FOLLOWING WORD]
[PARENT LENGTH]
[PARENT GENDER]
[PARENT NUMBER]
Features on the pair
[EXACT STRING MATCH (T/F)]
[HEAD MATCH (T/F)]
[SENTENCE DISTANCE, CAPPED AT 10]
[MENTION DISTANCE, CAPPED AT 10]

Table 1: Coreference features, associated to each candidate mention-mention arc in the tree. As in Durrett and Klein (2013), we also include conjunctions of each feature with the child and parent mention types (proper, nominal, or, if pronominal, the pronoun word).

Features on the quote-speaker pair
[WORD DISTANCE]
[SENTENCE DISTANCE]
[# IN-BETWEEN QUOTES]
[# IN-BETWEEN SPEAKERS]
[SPEAKER IN QUOTE, 1ST PERS. SG. PRONOUN (T/F)]
[SPEAKER IN QUOTE, 1ST PERS. PL. PRONOUN (T/F)]
[SPEAKER IN QUOTE, OTHER (T/F)]
Features on the speaker
[PREVIOUS WORD IS QUOTE (T/F)]
[PREVIOUS WORD IS SAME QUOTE (T/F)]
[PREVIOUS WORD IS ANOTHER QUOTE (T/F)]
[PREVIOUS WORD IS SPEAKER (T/F)]
[PREVIOUS WORD IS PUNCTUATION (T/F)]
[PREVIOUS WORD IS REPORTED SPEECH VERB (T/F)]
[PREVIOUS WORD IS VERB (T/F)]
[NEXT WORD IS QUOTE (T/F)]
[NEXT WORD IS SAME QUOTE (T/F)]
[NEXT WORD IS ANOTHER QUOTE (T/F)]
[NEXT WORD IS SPEAKER (T/F)]
[NEXT WORD IS PUNCTUATION (T/F)]
[NEXT WORD IS REPORTED SPEACH VERB (T/F)]
[NEXT WORD IS VERB (T/F)]

Table 2: Quotation attribution features, associated to each quote-speaker candidate. These features are used in the QUOTEONLY, QUOTEBEFORECOREF, and JOINT systems (where the speaker is a mention) and in the QUOTEAFTERCOREF system (where the speaker is an entity).

quotation-coreference tree can be obtained by first assigning the highest scored mention to each quotation, and then building a standard coreference tree involving only the mention nodes. This corresponds to the QUOTEBEFORECOREF baseline, to be used in §6.

To go beyond separate models, we introduce a final JOINT model, which includes additional scores that depend not just on arcs, but also on *paths* in the tree. Concretely, we select certain pairs of nodes and introduce scores for the event that both nodes are in the same branch of the tree. Rather than doing this for all pairs—which essentially would revert to the computationally demanding pairwise coreference models discussed in §3—we focus on a small set of pairs that are mostly related with the interaction between the two tasks we address jointly. Namely, we consider the mention-quotation pairs such that the mention

Mention-inside-quote features
[MENTION IS 1ST PERSON, SING. PRONOUN (T/F)]
[MENTION IS 1ST PERSON, PLUR. PRONOUN (T/F)]
[OTHER MENTION (T/F)]
Consecutive quote features
[DISTANCE IN NUMBER OF WORDS]
[DISTANCE IN NUMBER OF SENTENCES]

Table 3: Features used in the JOINT system for mention-quote pairs (only for mentions inside quotes) and for quote pairs (only for consecutive quotes). These features are associated to pairs in the same branch of the quotation-coreference tree.

span is within the quotation span (*mention-inside-quotation pairs*), and pairs of quotations that appear consecutively in the document (*consecutive-quotation pairs*). The idea is that, if consecutive quotations appear on the same branch of the tree, they will have the same speaker (the entity class associated with that branch), even though they are not necessarily siblings. These two pairs are aligned with the motivating examples (a) and (b) shown in §1.

4.2.1 Mention-inside-quotation features

The top rows of Table 3 show the features we defined for mentions inside quotes. The features indicate whether the mention is first-person singular pronominal (*I*, *me*, *my*, *myself*), which provides strong evidence that it co-refers with the quotation author, whether it is first-person plural pronominal (*we*, *us*, *our*, *ourselves*), which provides a weaker evidence (but sometimes works for colective entities that are organizations), and whether none of the above happens—in which case, the speaker is *unlikely* to be co-referent with the mention.

4.2.2 Consecutive quotation features

We show our consecutive quote features in the bottom rows of Table 3. We use only distance features, measuring both distance in sentences and in words, with binning. These simple features are enough to capture the trend of consecutive quotes that are close apart to have the same speaker.

5 Joint Decoding and Training

While decoding in the basic model is easy—as pointed out above, it can even be done by running a mention-level quotation attributor and the coreference resolver independently (QUOTEBEFORECOREF)—exact decoding with the JOINT model is in general intractable, since this model breaks the independence assumption between the arcs. However, given the relatively small amount of node pairs that have scores (only mentions inside quotations and consecutive quotations), we expect this "perturbation" to be small enough not to affect the quality of an approximate decoder. The situation resembles other problems in NLP, such as non-projective dependency parsing, which becomes intractable if higher order interactions between the arcs are considered, but can still be well approximated. Inspired by work in parsing (Martins et al., 2009) using linear relaxations with multi-commodity flow models, we propose a similar strategy by defining auxiliary variables and coupling them in a logic program.

5.1 Logic Formulation

We next derive the logic program for joint decoding of coreferences and quotations. The input is a set of nodes (including an artificial node), a set of candidate arcs with scores, and a set of node pairs with scores. To make the exposition lighter, we index nodes by integers (starting by the root node 0) and we do not distinguish between mention and quotation nodes. Only arcs from left to right are allowed. The variables in our logic program are:

- Arc variables $a_{i \rightarrow j}$, which take the value 1 if there is an arc from i to j, and 0 otherwise.
- Pair variables $p_{i,j}$, which indicate that nodes i and j are in the same branch of the tree.
- Path variables $\pi_{j \rightarrow^* k}$, indicating if there is a path from j to k.
- Common ancestor variables $\psi_{i \rightarrow^* j,k}$, indicating that node i is a common ancestor of nodes j and k in the tree.

Consistency among these variables is ensured by the following set of constraints:

- Each node except the root has exactly one parent:

$$\sum_{i=0}^{j-1} a_{i \rightarrow j} = 1, \ \forall j \neq 0 \qquad (1)$$

- There is a path from each node to itself:

$$\pi_{i \rightarrow^* i} = 1, \ \forall i \qquad (2)$$

- There is a path from i to k iff there is some j such that i is connected to j and there is path

from j to k:

$$\pi_{i \rightarrow^* k} = \bigvee_{i<j\leq k} (a_{i \rightarrow j} \wedge \pi_{j \rightarrow^* k}),\ \forall i, k \quad (3)$$

- Node i is a common ancestor of k and ℓ iff there is a path from i to k and from i to ℓ:

$$\psi_{i \rightarrow^* k,\ell} = \pi_{i \rightarrow^* k} \wedge \pi_{i \rightarrow^* \ell},\ \forall i, k, \ell \quad (4)$$

- Nodes k and ℓ are in the same branch if they have a common ancestor which is not the root:

$$p_{k,\ell} = \bigvee_{i \neq 0} \psi_{i \rightarrow^* k,\ell},\ \forall k, l. \quad (5)$$

The objective to optimize is linear in the arc and pair variables (hence the problem can be represented as an integer linear program by turning the logical constraints into linear inequalities).

5.2 Dual Decomposition

To decode, we employ the alternating directions dual decomposition algorithm (AD^3), which solves a relaxation of the ILP above. AD^3 has been used successfully in various NLP tasks, such as dependency parsing (Martins et al., 2011; Martins et al., 2013), semantic role labeling (Das et al., 2012), and compressive summarization (Almeida and Martins, 2013). At test time, if the solution is not integer, we apply a simple rounding procedure to obtain an actual tree: for each node j, obtain the antecedent (or root) i with the highest $a_{i \rightarrow j}$, solving ties arbitrarily.

5.3 Learning the Model

We train the joint model with the max-loss variant of the MIRA algorithm (Crammer et al., 2006), adapted to latent variables (we simply obtain the best tree consistent with the gold clustering at each step of MIRA, before doing cost-augmented decoding). The resulting algorithm is very similar to the latent perceptron algorithm in Fernandes et al. (2011), but it uses the aggressive stepsize of MIRA. We set the same costs for coreference mistakes as Durrett and Klein (2013), and a unit cost for missing the correct speaker of a quotation. For speeding up decoding, we first train a basic pruner for the coreference system (using only the features described in §4.1.1), limiting the number of candidate antecedents to 10, and discarding scores whose difference with respect to the best antecedent is below a threshold. We also freeze the best coreference trees consistent with the gold clustering using the pruner model, to eliminate the need of latent variables in the second stage.

6 Experiments

6.1 Dataset

We used the 597 documents of the Wall Street Journal (WSJ) corpus that were disclosed for the CoNLL-2011 coreference shared task (Pradhan et al., 2011) as a dataset for coreference resolution. This dataset includes train, development and test partitions, annotated with coreference information, as well as gold and automatically generated syntactic and semantic information.

The CoNLL-2011 corpus does not contain annotations of quotation attribution. For that reason, we used the WSJ quotation annotations in the PARC dataset (Pareti, 2012). We used the same version of the corpus as O'Keefe et al. (2012), but with different splits, to match the dataset partitions in the coreference resolution data. This attribution corpus contains 279 documents of the 597 CoNLL-2011 files, having a total of 1199 annotated quotes. As in that work, we only considered directed speech quotes and the direct part of mixed quotes (quotes with both direct and undirected speech).

6.2 Metrics for quotation attribution

Previous evaluations of quotation attribution systems were designed at *mention level*, and are thus assessed by comparing the predicted speaker mention span with the gold one. This metric assesses the amount of speaker mentions that were correctly identified. For compatibility with previous assessments, we report this score, which we call *Exact Match (EM)*: this is the percentage of predicted speakers with the same span as the gold one.

However, for several quotations (about 30% in the PARC corpus) this information is of little value, since the gold mention is a pronoun, which *per se* does not give any useful information about the actual speaker entity. Considering this fact, we propose two other metrics that capture information at the *entity level*, reflecting the amount of information a system is able to extract about the speakers:

- *Representative Speaker Match (RSM):* for each annotated quote, we obtain the full gold coreference set of the gold annotated speaker, and

choose a *representative speaker* from that cluster. We define this representative speaker as the proper mention which is the closest to the quote (if available); if the cluster does not contain proper mentions, we use the closest nominal mention; if only pronominal mentions are available, we use the original annotated speaker. The final measure is the percentage of predicted speakers that match the string of the corresponding representative speakers.

- *Entity Cluster F_1 (ECF_1).* Considering that a system outputs a set of mentions coreferent to the predicted speakers, we compute the F_1 score between the predicted set and the gold coreference cluster of the correct speaker.

The entity level metrics are not only useful for assessing the quality of an quotation attribution system—they also reflect the quality of the underlying coreference system used to cluster the related mentions.

6.3 Attribution baselines

To analyze the task of entity-level quotation attribution, we implemented three baseline systems.

- QUOTEONLY: A quotation attribution system trained on the representative speaker, instead of the gold speaker. For fairness, this baseline was trained with an extra feature indicating the type of the mention (nominal, pronominal or proper).
- QUOTEAFTERCOREF: An attribution system directly applied to the output of a predicted coreference chain. This baseline uses a coreference pre-processing, as applied in O'Keefe et al. (2012).
- QUOTEBEFORECOREF: An attribution system trained on the gold speaker, and post-combined with the output of a coreference system. This system should be able to provide a set of informative mentions about a quote, post-resolving the problem of the pronominal speakers. This kind of post-coreference approach was used by de La Clergerie et al. (2011).

6.4 Coreference Resolution

We use the coreference results of our basic QUOTEBEFORECOREF system as a baseline for coreference resolution. Since this system effectively solves the two problems separately, this can be considered our implementation of the SURFACE system of Durrett and Klein (2013) . As reported in Table 4, the perfromance of our baseline is comparable with the one of the SURFACE system of Durrett and Klein (2013), which is denoted as SURFACE-DK-2013.[2]

Table 4 also show the CoNLL metrics obtained for the proposed system of joint coreference resolution and quotation attribution. Our joint system outperformed the baseline with statistical significance (with $p < 0.05$ and according to a bootstrap resampling test (Koehn, 2004)) for all metrics expect for the CEAFE F_1 measure, whose value was only slighty improved. These results confirm that the coreference resolution task benefits for being tackled jointly with quotation attribution.

6.5 Quotation attribution

We implemented and trained the three attribution systems that were described in §6.3 and the system for joint coreference and author attribution that is detailed in §4. For each system, Table 5 shows the mention-based and entity-based metrics that were described in §6.2.

Training a quotation attribution system using representative speakers instead of the gold speakers (QUOTEONLY) leads to rather disappointing results. As expected, we conclude that assigning the semantically related speaker is considerably easier than selecting another mention that is coreferent with the correct speaker.

Using (predicted) coreference information, both QUOTEAFTERCOREF and QUOTEBEFORECOREF systems considerably increase our entity-based metrics. This was also expected, since the coreference chain allows these baselines to output a set of related mentions. We observed that, using the coreference resolution clusters as the attribution entity (QUOTEAFTERCOREF) influences the results negatively when compared to a more basic system that runs coreference on top of attribution result of the QUOTEONLY system (QUOTEBEFORECOREF). These results indicate that the quotation attribution task performs better by looking at the speaker mention that connects more strongly with the quotation, instead of trying to match the whole cluster.

Finally, the scores achieved by our JOINT

[2]To make the systems comparable, we re-trained Durrett et al.'s coreference system (version 0.9) on the WSJ portion of the Ontonotes datasets (the portion which has quote annotations from Pareti et al.'s PARC dataset). For this reason, the values in Table 4 differ from those reported in Durrett and Klein (2013), which where trained and tested in the entire Ontonotes.

	MUC F_1	BCUB F_1	CEAFE F_1	Avg.
SURFACE-DK-2013	**58.87**	62.74	45.46	55.7
SURFACE-OURS [QUOTEBEFORECOREF]	57.89	62.50	45.48	55.3
JOINT	58.78	**63.79**	**45.50**	**56.0**

Table 4: Coreference obtained with the CoNLL scorer (version 5) in the test partition of the WJS corpus, for the SURFACE system of Durrett and Klein (2013), our baseline implementation of the that system (SURFACE-OURS), and our JOINT approach. All systems were trained in the WSJ portion of the Ontonotes.

	EM	RSM	ECF_1
QUOTEONLY	49.1%	49.4%	41.2%
QUOTEAFTERCOREF	76.7%	64.6%	70.0%
QUOTEBEFORECOREF	**88.7%**	74.7%	73.7%
JOINT	88.1%	**76.6%**	**74.1%**

Table 5: Attribution results obtained, in the test set, for the three baseline systems and our joint system.

model are slightly above the best baseline system QUOTEBEFORECOREF, yielding the best performance on the entity-level quotation attribution task. The differences, however, were not found statistically significant, probably due to the small number of quotes (159) in the test set.

The average decoding runtime of the JOINT model is 1.6 sec. per document, against 0.2 sec. for the pipeline system. This slowdown is expectable given the fact that the pipeline system only needs to make independent decisions, while the joint version needs to solve a harder combinatorial problem. Yet, this runtime is within the order of magnitude of the time necessary to preprocess the documents (which includes tagging and parsing the sentences).

6.6 Error Analysis

To understand the type of errors that are prevented with the JOINT system, consider the following example (from document WSJ-2428):

- [Robert Dow, a partner and portfolio manager at Lord, Abbett & Co.]$_{M_1}$, which manages $4 billion of high-yield bonds, says [he]$_{M_2}$ doesn't *"think there is any fundamental economic rationale (for the junk bond rout). It was [herd instinct]*$_{M_3}$*."* [He]$_{M_4}$ adds: *"The junk market has witnessed some trouble and now some people think that if the equity market gets creamed that means the economy will be terrible and that's bad for junk."*

The basic QUOTEBEFORECOREF system wrongly clusters together M_3 and M_4 as coreferent, and wrongly assigns M_3 as the representative speaker. On the other hand, the JOINT system correctly clusters M_1, M_2 and M_4 as coreferent. This is due to the presence of the consecutive quote features which aid in understanding that both quotes have the same speaker, and the mention-inside-quote features which prevent *herd instinct*, which is inside a quote, from being coreferent with *He*, which is very likely the author of the quotes due to the verb *adds*.

7 Conclusions

We presented a framework for joint coreference resolution and quotation attribution. We represented the problem as finding an optimal spanning tree in a graph including both quotation nodes and mention nodes. To couple the two tasks, we introduce variables that look at paths in the tree, indicating if pairs of nodes are in the same branch, and we formulate decoding as a logic program. Each branch from the root can then be interpreted as a cluster containing all coreferent mentions of an entity and all quotes from that entity.

In addition, we designed an evaluation metric suitable for entity-level quotation attribution that takes into account informative speakers. Experimental results show mutual improvements in the coreference resolution and quotation attribution tasks.

Future work will include extensions to tackle indirect quotations, possibly exploring connections to semantic role labeling.

Acknowledgements

We thank all reviewers for their valuable comments, and Silvia Pareti and Tim O'Keefe for providing us the PARC dataset and answering several questions. This work was partially supported by the EU/FEDER programme, QREN/POR Lisboa (Portugal), under the Intelligo project (contract 2012/24803) and by a FCT grant PTDC/EEI-SII/2312/2012.

References

M. B. Almeida and A. F. T. Martins. 2013. Fast and robust compressive summarization with dual decomposition and multi-task learning. In *Proc. of the Annual Meeting of the Association for Computational Linguistics*.

Eric Bengtson and Dan Roth. 2008. Understanding the value of features for coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 294–303. Association for Computational Linguistics.

Shane Bergsma and Dekang Lin. 2006. Bootstrapping path-based pronoun resolution. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 33–40. Association for Computational Linguistics.

K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. 2006. Online Passive-Aggressive Algorithms. *Journal of Machine Learning Research*, 7:551–585.

Aron Culotta, Michael Wick, Robert Hall, and Andrew McCallum. 2007. First-order probabilistic models for coreference resolution. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pages 81–88.

D. Das, A. F. T. Martins, and N. A. Smith. 2012. An Exact Dual Decomposition Algorithm for Shallow Semantic Parsing with Constraints. In *Proc. of First Joint Conference on Lexical and Computational Semantics (*SEM)*.

Éric de La Clergerie, Benoît Sagot, Rosa Stern, Pascal Denis, Gaëlle Recourcé, and Victor Mignot. 2011. Extracting and visualizing quotations from news wires. In *Human Language Technology. Challenges for Computer Science and Linguistics*, pages 522–532. Springer.

Pascal Denis and Jason Baldridge. 2008. Specialized models and ranking for coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 660–669. Association for Computational Linguistics.

Greg Durrett and Dan Klein. 2013. Easy victories and uphill battles in coreference resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*.

Greg Durrett, David Hall, and Dan Klein. 2013. Decentralized entity-level modeling for coreference resolution. In *Proc. of Annual Meeting of the Association for Computational Linguistics*.

David K Elson and Kathleen McKeown. 2010. Automatic attribution of quoted speech in literary narrative. In *AAAI*.

William Paulo Ducca Fernandes, Eduardo Motta, and Ruy Luiz Milidiú. 2011. Quotation extraction for portuguese. In *Proceedings of the 8th Brazilian Symposium in Information and Human Language Technology (STIL 2011), Cuiabá*, pages 204–208.

Eraldo Rezende Fernandes, Cícero Nogueira dos Santos, and Ruy Luiz Milidiú. 2012. Latent structure perceptron with feature induction for unrestricted coreference resolution. In *Joint Conference on EMNLP and CoNLL-Shared Task*, pages 41–48. Association for Computational Linguistics.

J.R. Finkel, A. Kleeman, and C.D. Manning. 2008. Efficient, feature-based, conditional random field parsing. *Proc. of Annual Meeting on Association for Computational Linguistics*, pages 959–967.

Aria Haghighi and Dan Klein. 2007. Unsupervised coreference resolution in a nonparametric bayesian model. In *Annual meeting-Association for Computational Linguistics*, volume 45, page 848.

Aria Haghighi and Dan Klein. 2010. Coreference resolution in a modular, entity-centered model. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 385–393. Association for Computational Linguistics.

P. Koehn. 2004. Statistical signicance tests for machine translation evaluation. In *Proc. of the Annual Meeting of the Association for Computational Linguistics*.

Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 28–34. Association for Computational Linguistics.

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

A. F. T. Martins, N. A. Smith, and E. P. Xing. 2009. Concise Integer Linear Programming Formulations for Dependency Parsing. In *Proc. of Annual Meeting of the Association for Computational Linguistics*.

A. F. T. Martins, N. A. Smith, P. M. Q. Aguiar, and M. A. T. Figueiredo. 2011. Dual Decomposition with Many Overlapping Components. In *Proc. of Empirical Methods for Natural Language Processing*.

A. F. T. Martins, M. B. Almeida, and N. A. Smith. 2013. Turning on the turbo: Fast third-order non-projective turbo parsers. In *Proc. of the Annual Meeting of the Association for Computational Linguistics*.

Vincent Ng and Claire Cardie. 2002. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 104–111. Association for Computational Linguistics.

V. Ng. 2010. Supervised noun phrase coreference research: The first fifteen years. In *Proc. of the Annual Meeting of the Association for Computational Linguistics*.

Tim O'Keefe, Silvia Pareti, James R Curran, Irena Koprinska, and Matthew Honnibal. 2012. A sequence labelling approach to quote attribution. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 790–799. Association for Computational Linguistics.

Silvia Pareti, Tim O'Keefe, Ioannis Konstas, James R. Curran, and Irena Koprinska. 2013. Automatically detecting and attributing indirect quotations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*.

Silvia Pareti. 2012. A database of attribution relations. In *LREC*, 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.

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. 2011. Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–27. Association for Computational Linguistics.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Proceedings of the Joint Conference on EMNLP and CoNLL: Shared Task*, pages 1–40.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K Joshi, and Bonnie L Webber. 2008. The penn discourse treebank 2.0. In *LREC*. Citeseer.

Altaf Rahman and Vincent Ng. 2011. Narrowing the modeling gap: A cluster-ranking approach to coreference resolution. *Journal of Artificial Intelligence Research*, 40(1):469–521.

Luis Sarmento, Sergio Nunes, and E Oliveira. 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, Frederick L Crabbe, and Noah A Smith. 2010. Visualizing topical quotations over time to understand news discourse. Technical report, Technical Report CMU-LTI-01-103, CMU.

Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational linguistics*, 27(4):521–544.

Veselin Stoyanov, Nathan Gilbert, Claire Cardie, and Ellen Riloff. 2009. Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 656–664. Association for Computational Linguistics.

Yannick Versley, Simone Paolo Ponzetto, Massimo Poesio, Vladimir Eidelman, Alan Jern, Jason Smith, Xiaofeng Yang, and Alessandro Moschitti. 2008. Bart: A modular toolkit for coreference resolution. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Demo Session*, pages 9–12. Association for Computational Linguistics.