A Joint Inference Architecture for Global Coreference Clustering with Anaphoricity

Thomas Bögel and Anette Frank

Department of Computational Linguistics
Heidelberg University
{boegel,frank}@cl.uni-heidelberg.de

Abstract. We present an architecture for coreference resolution based on joint inference over anaphoricity and coreference, using Markov Logic Networks. Mentions are discriminatively clustered with discourse entities established by an anaphoricity classifier. Our entity-based coreference architecture is realized in a joint inference setting to compensate for erroneous anaphoricity classifications and avoids local coreference misclassifications through global consistency constraints. Defining pairwise coreference features in a global setting achieves an efficient entity-based perspective. With a small feature set we obtain a performance of 63.56% (gold mentions) on the official CoNLL 2012 data set.

1 Introduction

Coreference resolution (CR) is the task of detecting and clustering mentions of discourse entities in a text. Successfully resolving coreferences is an important and challenging step in many NLP tasks. This paper takes a discourse-oriented perspective on CR, by realizing a global, entity-centric approach for mention clustering that exploits dependencies between anaphoricity and coreference. Often, clustering is done after pairwise CR to resolve inconsistencies. We are using a joint approach that performs CR classification and clustering simultaneously, while imposing global consistency constraints. To reduce the complexity of this global clustering approach, we use joint inference with the results of an anaphoricity classifier. Using the anaphoricity classifier, we create anchors for discourse entities and cluster mentions with those entities. Treating both processes in a joint inference architecture ensures that errors of the anaphoricity classifier can be counterbalanced, thus avoiding pipeline effects. Global consistency constraints and exploitation of pairwise features in a global, entity-based coreference formalization enable us to solve the problem efficiently.

Our architecture is based on Markov Logic Networks (Richardson and Domingos, 2006) which combine first-order logic formulas with a probabilistic model. This allows for transparent formalization of the architecture and of the interaction constraints, as well as flexible extensions of the feature set.

This paper is organized as follows: Sec. 2 presents related work. Sec. 3 describes our core architecture for global coreference clustering with anaphoricity. Sec. 4 introduces the components for anaphoricity and coreference determination. Sec. 5 presents our experiments and results and Sec. 6 concludes.

I. Gurevych, C. Biemann, and T. Zesch (Eds.): GSCL 2013, LNAI 8105, pp. 35–46, 2013.

2 Modeling Coreference Resolution

Approaches to Coreference Resolution. Machine learning approaches have long addressed CR in a pairwise scenario deciding for each pair of mentions whether they are coreferent or not. A subsequent clustering step then clusters the pairs to entities (Soon et al. (2001), Ng and Cardie (2002)). This is often implemented in a pipeline architecture and thus suffers from error propagation and locality issues, such as violation of transitivity over local CR decisions.

Entity-based approaches, on the other hand, focus on modeling the entity itself by using entity-level features. This reduces the problem of locality as a mention is compared to a complete entity instead of creating an entity by pairwise classification decisions. Defining entity-level features is challenging, as they require appropriate representations for judging similarity between mentions and an entity. Features for entities can, for instance, be defined by aggregating and comparing specific attributes for all mentions in an entity (Wellner et al., 2004).

A further challenge of entity-based approaches is the number of clusters to be examined. Luo et al. (2004) make use of Bell trees to reduce this processing complexity. Martschat et al. (2012) perform global CR using multigraphs, allowing for an entity-based perspective. Finally, ranking approaches are used to find the best antecedent entity for mentions (Rahman & Ng (2011); Denis & Baldridge (2008)) instead of relying on single pairwise classifications.

Joint Inference for Coreference Resolution. To overcome the problem of locality, different aspects of CR can be combined using joint inference techniques. Denis and Baldridge (2009) use Integer Linear Programming to perform joint inference over independent classifier decisions for anaphoricity, pairwise coreference and named entity type. They define global ILP constraints over these classifications to determine a globally optimal solution that respects dependencies of anaphoricity and coreference decisions. In contrast to our approach, pairwise coreference decisions need to be harmonized by explicit transitivity constraints. These could not be fully implemented due to efficiency constraints. Recently, Song et al. (2012) proposed a joint inference formalization using MLN that interfaces local, pairwise resolution and clustering by way of explicit transitivity constraints. We, in contrast, will use anaphoricity as an anchor for entities and perform best-first clustering without stating explicit transitivity constraints.

Poon and Domingos (2008) use MLN to perform unsupervised CR in an entity-based approach that, like ours, implicitly accounts for transitivity. Their formalization does not model interactions with anaphoricity and offers a restricted set of entity-level features. Clustering of mentions is driven by head features, and few semantic type and morphological features are used to assign further mentions to these clusters. Other factors, such as distance, are encoded using a pre-defined prior. Thus their formalization is not truly transparent from a linguistic modeling point of view. As many rules are defined as hard constraints, it is unclear whether the system can be adapted to a supervised scenario, in order to adapt to specific domains, and whether an extended set of features can be integrated using hard constraints or independently determined priors.

2.1 Markov Logic Networks

Markov Logic Networks (MLN) combine two major advantages desirable for NLP applications: (1) being able to model uncertainty efficiently (using probabilistic graphical models) and (2) expressing and combining various sources of knowledge (by first-order logic) (Richardson and Domingos, 2006).

A MLN consists of a predicate logic knowledge base (formulas) with a weight attached to each formula. The weight of a formula can be regarded as the cost of violating it. A high accumulated cost over all formulas leads to a less probable possible world and vice versa. More formally, the probability of a specific state of a world x can be described in a log-linear model as follows:

$$P(X = x) = \frac{1}{Z} \exp(\sum_i w_i n_i(x)) \tag{1}$$

The weight w_i of each formula is multiplied by the number of true groundings of the formula in the specified world $(n_i(x))$.

Each MLN instantiates a Markov network: binary nodes in the network correspond to possible groundings of first-order predicates and one feature (edge) for each formula with the corresponding weight. Each set of ground atoms from a knowledge base thus leads to a specific ground Markov network.

Learning. Learning weights discriminatively for a specific predicate is a problem of maximizing the conditional probability of a query given a possible world: $P(Y = y|X = x) = \frac{1}{Z_x} \exp \sum_i w_i n_i(x, y)$, where x are evidence predicates and y the query predicate. $n_i(x, y)$ counts the number of true groundings of the ith formula. Weights are learned to maximize the conditional probability. Intuitively, the number of true groundings in the data is compared to the expected number of true groundings. With a gradient descent method, this problem can be solved efficiently (Lowd and Domingos, 2007)

Inference. MAP inference aims at finding the most probable state y (query predicate) for given evidence x ($\arg\max_y P(y|x)$), which can be expressed as $\sum_i w_i n_i(x, y)$. Satisfiability solvers solve this NP-hard problem efficiently.

We use Tuffy (Niu et al., 2011) as an inference framework for MLN due to its efficiency during inference. Tuffy implements the Newton Diagonal method for learning weights and MaxWalkSAT for inference.

3 Global CR Clustering with Anaphoricity Using MLN

Our approach aims at modeling CR using a *global, entity-based approach* that will be complemented by a *discourse-based perspective*, by using *discourse-new*, or *anaphoricity* information to guide the entity clustering process. This architecture will be formulated as a joint inference problem using MLN.

```
1 // predicate declarations
2 *cand_ent(MEN,ENT)
3 m_in_e(MEN,ENT!)
4 *anaph(MEN,BOOLEAN)
5 *feat(MEN,FEAT)

6 //discourse-new mentions initiate an entity
7 cand_ent(m,e) , anaph(m,0) ⇒ m_in_e(m,e)
8 cand_ent(m,e) , anaph(m,1) ⇒ !m_in_e(m,e)

9 // rule schema for pairwise CR clustering
10 m_in_e(m_e,e), !m_in_e(m,e) , m != m_e,
   feat(m_e,m,val) ⇒ m_in_e(m,e)
11 // constraint: resolve all mentions
12 cand_ent(m,e) ⇒ m_in_e(m,e_x) .
```

Fig. 1. Core predicate declarations and rules. , indicates conjunction, ⇒ a conditional and ! negation. $*$ specifies a closed-world assumption and . defines a hard constraint.

We realize a global, entity-based CR system by clustering all mentions into entities and choosing a partition of entities that simultaneously satisfies all coreference features. Such an approach circumvents the problem of local classifications. Implementing this in a brute-force manner is computationally prohibitive, as it requires considering the probability of each possible clustering of mentions. As $\mathcal{P}(M) = 2^n$ for a set M of n mentions, the number of possibilities to compute increases exponentially with the number of mentions. Luo et al. (2004) introduce a bell tree to efficiently manage multiple clusterings of mentions using a beam search, but pruning bears the risk of removing globally good results.

Instead of considering all possible clusterings, we will use anaphoricity as a guide to establish discourse entities for subsequent mention clustering. Mentions that are determined by the anaphoricity classifier to be *non-anaphoric*, or *discourse-new*, will be considered as an 'anchor' for a new discourse entity that serves as a reference point for discriminative clustering of mentions classified as *anaphoric*. Combining anaphoricity and CR in a joint inference architecture with soft constraints allows us to compensate for errors in either module. In contrast to Denis and Baldridge (2009), where pairwise classifications are supported by anaphoricity, our clustering approach focuses on entities from the beginning.

We use Markov Logic Networks, as they offer a flexible and transparent framework that can be easily extended to incorporate additional knowledge.

3.1 Global Architecture and MLN Formalization

Our MLN formalization consists of four main components. We first give a brief overview of these components and their interplay and then turn to discuss them in more detail. Fig. 1 states relevant predicate declarations and rules.

1. **Initialization.** *All* mentions m are declared as potential anchors for an entity e, through a predicate `cand_ent(MEN,ENT)` (l. 2) and appropriately defined knowledge base entries.
2. **Establishing Entities through Anaphoricity.** Mentions m classified as discourse-new by an external classifier instantiate a unique discourse entity e (the one associated with m through `cand_ent(m,e)`), through the coreference

indicating predicate `m_in_e(m,e)`. (l. 3, 7). Mentions classified as anaphoric do *not* initiate a discourse entity (l. 8).

3. Global Coreference Resolution. The rule schema in line 10 applies to *all* mentions m that are not (yet) clustered with an entity e, and evaluates the strength of individual CR features *feat* (l. 5) holding between m and any distinct mention m_e clustered with e.This implements an entity-based CR strategy while using mention-level features. The rule quantifies over all entities e that were established by a discourse-new mention in **2.** (l. 7).

4. Two Constraints ensure consistency and constrain the clustering of mentions to entities:

a. Uniqueness: *Each mention is assigned to exactly one entity.* This constraint (defined through `m_in_e(MEN,ENT!)`, l. 3) implies *disjointness of discourse entities.* It considerably reduces the combinatorics of clusterings, since anchor mentions m_e of entities e need not be considered by the coreference clustering rules in **3**: a mention m_e that is already clustered with an entity e cannot be clustered with a disjoint entity $e_x \neq e$.

b. Resolve All Mentions. This constraint (l. 12) enforces that each mention is clustered with an entity: the entity of which it is an anchor ($e = e_x$) or some other entity ($e \neq e_x$). When resolving gold mentions, we define it as a hard constraint. It can be relaxed when dealing with system mentions.

Transitivity. Unlike Denis&Baldridge (2009) and Song et al. (2012) we do not encode explicit transitivity constraints. Violations of transitivity are excluded by ***Uniqueness***. Since clustering is entity-centric (through `m_in_e(m,e)`), the uniqueness constraint implies *disjointness* of entities, and thus, discriminative clustering of mentions to entities.[1]

3.2 Using Anaphoricity to Establish Entities

Anaphoricity determines whether a mention is anaphoric (i.e., refers to a previously mentioned entity) or non-anaphoric and thus introduces a new discourse entity. Mentions classified as discourse-new can be used as 'seeds' for establishing entities for subsequent (discriminative) clustering of anaphoric mentions to the created discourse entities. We exploit linguistic knowledge about anaphoricity to establish the set of discourse entities in a given text, and thereby considerably reduce the space of possibilities to be investigated for entity clustering.

Anaphoricity of mentions is determined by an external anaphoricity classifier (AC) (see Sec. 4.1) that provides binary anaphoricity labels (l. 7, 8). Rule 7 receives high weight with mentions classified as *discourse-new (non-anaphoric)*, and instantiates a discourse entity, of which it is typically the first member. Rule 8 applies to the complementary case, with mentions classified as *anaphoric*. Here, the rule states that m does not instantiate a discourse entity.

Reducing Complexity. Our approach provides an initial partitioning of mentions into a set of anchored discourse entities. Since *uniqueness* forces each mention

[1] A sufficiently high weight of the AC rule avoids clustering all mentions into a single entity, as it triggers the creation of a sufficient number of entities.

to be contained in a single entity, mentions serving as anchors for an entity cannot be clustered with other entities. This knowledge about disjointness severely reduces the combinatorics of coreference-based clustering of mentions to entities.

Avoiding Pipeline Effects. While our approach reduces the number of clusterings to be examined, errors of the AC may severely harm CR performance (Denis and Baldridge, 2009). To avoid pipeline effects, the rules that create discourse entities are weighted to counterbalance errors of the AC: We allow for *each* mention to instantiate a new entity with a learned weight that depends on the AC result. At present, we use two rules (l. 7, 8) to represent the binary AC predictions.

3.3 Global Coreference Resolution

There are different ways for using CR features in an entity-based architecture: *cluster- or entity-level features*, i.e., features defined between a mention and an entity, or *mention-level features*, as typically used in *mention-pair* models. As the design of entity-level features is challenging for some feature types (e.g. distance measures) and computing feature values for all possible entity clusterings is computationally expensive, we make use of *mention-level features* that are evaluated in an *entity-centric*, *discriminative ranking approach* by exploiting universal quantification of variables in MLN formulae (Fig. 1, l. 10).

A mention m that is to be classified will be evaluated for coreference with every entity e by comparing it to all mentions in the cluster and accumulating the evidence for a mention to be in a specific cluster using the rule weights for individual pairwise coreference features. In this way, we compare a mention to *all* other mentions of the entity using pairwise, mention-level features.

In MLN, this intuition is captured by exploiting implicitly universally quantified variables. I.e., for each feature $feat$ and possible value val of $feat$, we employ the rule schema from Fig. 1, l. 10: For all mentions m_e in an entity e: if we observe a feature value for the mention pair (m_e, m), m should also be assigned to e (with rule weight w_{feat}). For features encoding negative evidence, the learned rule weight is negative. According to equation (1), the weight for all instantiated formulas are accumulated. The final weight for assigning mention m to some entity e is thus obtained by comparing m to *all* mentions m_e in cluster e, and we assign m to the entity that receives the highest overall score.

3.4 Joint Inference

We now discuss how the components for anaphoricity and coreference interact to allow for joint inference and ensure consistency for the entire inference process.

Anaphoricity and Coreference. Anaphoricity information is used to provide seeds for entity clusterings to which anaphoric mentions are attached. Misclassifications of the AC need to be counterbalanced by coreference features.

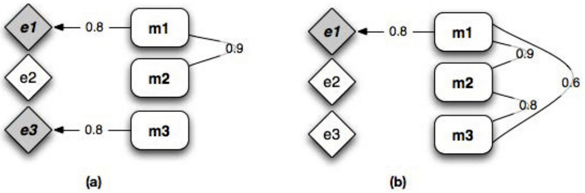

Fig. 2. Joint inference: anaphoricity and coreference

Fig. 2 demonstrates a relevant case: We assume two possible inference results (a) and (b) for a hypothetical document that contains three mentions ($\{m_1, m_2, m_3\}$). Each mention could instantiate a new entity ($\{e_1, e_2, e_3\}$). The AC predicts that mentions m_1 and m_3 are discourse-new which results in a high weight (0.8, in this example) for the mentions to instantiate a new entity (Fig. 1, l. 7). We assume that m_3 is *erroneously* classified as discourse-new. Solution (a) accepts the AC prediction and clusters mentions m_1 and m_2 with e_1 and m_3 as a new entity e_3, whereas solution (b) rejects the AC prediction for m_3.

In addition to anaphoricity, there is an accumulated weight due to all *coreference* indicators between each pair of mentions. For solution (a), the accumulated scores yield an overall weight of $0.8 + 0.8 + 0.9 = 2.7$. Solution (b) rejects the result of the AC and instead attaches m_3 to entity e_1. The overall score of $0.8 + 0.8 + 0.9 + 0.6 = 3.3$ exceeds solution (a). Thus, strong coreference features can indeed override errors of anaphoricity classification.

Conversely, features that contribute strong negative evidence for coreference can help to compensate for discourse-new mentions that are wrongly predicted to be anaphoric. If a mention does not fit with any cluster, it is likely to be non-anaphoric and to instantiate a new entity. Indeed, given strong negative evidence for coreference with any entity, we might obtain a globally optimal solution that establishes an entity using the fallback rule of line 12.

Modeling the mutual interdependence of coreference and anaphoricity decisions using joint inference offers a great advantage over pipeline architectures.

Interaction between Coreference Features. The proposed architecture also handles interactions and contradictions between coreference features. Each mention is evaluated by multiple coreference rules, each one defining different features and values, with different rule weights attached to them. As all rule weights for all features are accumulated (cf. equation (1)), positive or negative evidence for clustering a mention with an entity, are balanced against each other.

4 Anaphoricity and Coreference Features

4.1 Anaphoricity Classifier

We follow the evaluations in Poesio et al. (2005) and selected the most promising features from previous works. We optimized classification performance on the development set of the CoNLL 2012 dataset and chose the Random Forest classifier implemented in Weka (Hall et al., 2009) which yielded highest performance.

For each mention to be classified we determine its mention type, whether it occurs in the first sentence or is the first occurrence of the head or surface form of the mention. We check for pre- and post-modification, definiteness, superlative forms and the grammatical function. In addition to these classical features, we use 8 measures to capture a raise in term frequency and tf-idf after the first mention of an entity that also holds for partial string matches (Ritz, 2010).

Training the anaphoricity classifier on the complete training portion of the CoNLL 2012 data and evaluating it on the development set yields an accuracy of 86.38% (Prec.: 86.5%, Rec.: 86.4%).

4.2 Coreference Features

We selected and implemented 17 features for coreference resolution from (Bengtson and Roth, 2008) and used them to test our architecture.[2] For each feature and possible feature value, we add a dedicated rule. For continuous features (e.g. distances), we first learn weights for each possible feature value individually and subsume values with a similar weight to obtain plausible feature ranges. We re-estimate weights for the obtained feature ranges.

As a fallback, we add a feature that attaches mentions to the *nearest potential antecedent* if there is not enough evidence for coreference. This avoids promoting unbound anaphoric mentions to independent discourse entities.

5 Feature Selection and Experiments

5.1 Experimental Setting

We use CoNLL 2012 Shared Task data (Pradhan et al., 2012) for all experiments and evaluate on the official test set. As our aim in this work is to develop a core baseline architecture as a proof of concept, we focus on *gold standard mentions.* Future work will extend our architecture to include system mentions.

We apply the five evaluation metrics used in the CoNLL 2012 Shared Task: MUC , B^3, CEAF with both the entity- and the mention-based similarity metric and BLANC. The arithmetic mean of all five F_1 scores is used for feature selection and presentation of results.

5.2 Weight Learning and Feature Selection

Despite using pairwise rules, learning weights for many or all rules simultaneously is still computationally expensive: each predicate that is added to the rule file adds a node and edges for each possible grounding of the formula to the Markov

[2] Surface features (*HeadMatch, StringMatch, Alias, StringKernelSim.*); Syntactic (*Appositive, Predicative, BindingConstraints, HobbsDistance*); Semantic (*Synonymy, Antonymy, SemanticDistance*); Agreement (*Gend-/Num-/Semantic-Agr*); Distance (*Token-/Mention-/Sent-Dist*).

Table 1. Performance impact of features in additive feature selection: difference relative to last iteration ($\Delta avg(F)$) and absolute performance ($avg(F)$)

HeadMatch	-	59.82
Added rule	$\Delta avg(F)$	$avg(F)$
STRING KERNEL SIMILARITY	+0.56	60.38
GENDER AGREEMENT	+0.20	60.58
NEAREST ANTECEDENT	+0.04	60.62
HOBBS DISTANCE	+1.68	62.30
STRING MATCH	+1.64	63.94
REFLEXIVE MATCH	+1.35	65.29
SEMANTIC DISTANCE	+0.00	**65.29**

network. Dependencies between features in particular result in an exponential growth of the Markov network.

To reduce the size of the resulting Markov network and speed up the learning process, we learn weights for different rules (i.e. coreference feature values) individually. That is, we make a simplifying independence assumption *for all rules* so that we can learn rule weights individually: for independent features, each rule only contains one feature value. Nevertheless, we add the mention type to restrict rules to mentions for which a feature is appropriate. All weights are learned on 50 randomly sampled documents of the CoNLL 2012 training set, containing 2279 mentions in 513 entities.

We extracted features based on the provided automatic annotations and used the output of the AC during training, in order to ensure that the influence of erroneous anaphoricity annotation is learned and counterbalanced.

Additive Feature Selection. To determine an optimal feature set, we conduct greedy forward selection in combination with step-wise backward deletion. We start with a rule set without any coreference features (l. 1-8, 12 in Fig. 1) and add individual coreference rules one at a time to determine which rule yields highest overall performance gain (CoNLL score). This rule is then added to the rule set and the process is repeated by adding further rules until no further improvements are observed. After this process, we perform step-wise backward deletion: at each step, we eliminate one feature. If performance increases after deletion of a single feature, the feature is removed and we continue with forward selection again. This combination of forward selection and backward deletion is repeated until no improvements are observed.

Table 1 lists the features that were selected during the described process, and their contribution to overall performance on the development set.

5.3 Experiments and Results

Evaluation Setup. For final evaluation, we measure CR performance with the selected feature set (Table 1) on the test set using gold mentions and automatically created linguistic annotations.

Table 2. Evaluation results on the test set for four scenarios with optimized features

Evaluation Scenario	**(I) AC$_{auto}$ learned weight**				**(II) AC$_{auto}$ constraint**	**(III) AC$_{gold}$ learned weight**		**(IV) AC$_{gold}$ constraint**	
documents	*all*			*non-split*	*all*	*all*	*non-split*	*all*	*non-split*
	Prec	Rec	F$_1$	F$_1$	F$_1$	F$_1$	F$_1$	F$_1$	F$_1$
MUC	72.55	76.14	74.30	76.31	69.80	75.10	77.08	75.02	77.13
B^3	61.03	67.05	63.90	64.28	58.28	64.20	65.71	64.98	67.31
CEAF-M	64.14	86.54	66.27	66.42	53.26	66.58	66.99	67.47	69.26
CEAF-E	55.47	62.92	58.96	59.92	47.55	59.21	62.93	60.19	62.14
BLANC	52.71	56.14	54.37	60.76	53.21	54.64	62.28	55.00	63.63
CoNLL			**63.56**	65.54	56.42	63.95	67.00	64.53	67.90

Next to the full architecture with joint inference over anaphoricity (AC) and CR classifications (**I**) we evaluate further system variants to highlight the impact of joint inference and the individual anaphoricity and coreference sub modules (cf. Table 2): (**II**) highlights the effect of joint inference against a pipeline of *automatic* AC predictions and a *hard constraint* for establishing entities;[3] (**III**) and (**IV**) use oracle (*gold*) AC results with *learned weights* vs. *hard constraints* for the creation of entities. They illustrate the upper bound of the system's current AC integration and CR clustering performance.

In the CoNLL data, long documents are split into multiple pieces which artificially creates new entities at each break. As our anaphoricity-driven architecture is heavily influenced by such noise, we additionally evaluate most scenarios on the subset of documents that are not affected by such splits (*non-split*).

Results. Our final evaluation results are given in (**I**). With **63.56%**, the F$_1$-score of our model lies within the range of published results for the CoNLL 2012 Shared Task with gold mentions, where scores range from 51.40% to 77.22%.[4] A pipeline architecture (**II**) suffers from a drop of around 7 percentage points. This clearly shows that our joint inference architecture is effective in counterbalancing AC errors. We note a strong effect for both CEAF metrics.

Scenarios (**III**) and (**IV**) mark upper bounds for our approach regarding AC integration and CR performance. As our AC classifier scores at 81.6/84.7/83.12 P/R/F$_1$ on the test set, the small performance difference (0.39 points F$_1$) between automatic (**I**) and gold AC using learned rule weights (**III**) shows that the learned rule weights are well set. The results using an AC oracle with hard constraints for entity creation shows small differences, too. This points to a weakness of the current model regarding the performance of CR features.

For all models we observe clear performance increases for non-split documents, which avoids artificial noise that the AC is unable to detect.

[3] We simulate a pipeline by marking both anaphoricity rules as hard constraints.

[4] `http://conll.cemantix.org/2012/`

5.4 Error Analysis and Discussion

Errors involving anaphoricity. Joint inference over anaphoricity and coreference is crucial to our architecture. We thus measured how well joint inference counterbalances **(1)** *false positives (FP)* (mentions erroneously classified as anaphoric) and **(2)** *false negatives (FN)* (anaphoric mentions classified as non-anaphoric).

On the test set, the anaphoricity classifier yields a precision of 81.6% and a recall of 84.7%. In setting **I**, 32% of the FPs and 68% of the FNs are corrected. Thus, errors introduced by the AC erroneously classifying a mention as anaphoric are harder to resolve. Inspection of errors reveals that 52% of the mentions that are not corrected in our joint inference scenario (**I**) are pronouns. Most of these erroneously classified pronouns behave as discourse-new in the gold standard due to the fact that long documents are split into pieces in the CoNLL data set, which artificially creates new entities. As our AC is based on linguistic features, it is deemed to misclassify these mentions as anaphoric. Our approach is especially sensitive to this artificial noise as it depends on correct anaphoricity information. If we remove split files, 43% of FPs and 65% of FNs are corrected.

For FNs, stronger CR indicators are needed such that mentions could be attached to other entities despite being classified as discourse-new.

Discussion and Future Extensions. Our evaluation clearly shows that joint inference over anaphoricity is well designed, so that CR information can counterbalance classifier mistakes. Further extensions will integrate classifier confidence values, to help the impact of CR features for correcting false negatives.

The small feature set we are currently using shows that the architecture itself plus some strong features result in a strong baseline system. At the same time, the evaluation points to weaknesses of our current CR feature set. This is (partially) due to the quite strong independence assumptions during learning. In current work we perform feature selection using linguistically motivated feature groups and also use larger training sets, using a more efficient MLN engine.

6 Conclusion

In this paper we propose an architecture for CR that uses anaphoricity to establish a set of discourse entities in a text and clusters all anaphoric mentions with these entities. By accumulating weights of pairwise mention-level coreference comparisons we realize discriminative mention clustering while circumventing the problem of defining entity-level features. We use Markov Logic Networks as a framework to perform joint inference over the output of an anaphoricity classifier and pairwise entity-centric coreference decisions, and show that the system is able to correct errors of both anaphoricity and coreference. To our knowledge, this is the first attempt to use anaphoricity to establish discourse entities for discriminative global mention clustering. This is what clearly distinguishes our account from Poon and Domingos (2008), where discourse entities are globally clustered by the heads of mentions and agreement features. Our system achieves good performance using a small feature set. We are currently working with feature combinations on larger training set sizes and integrate system mentions to realize a powerful end-to-end system. First experiments yield promising results.

References

Bengtson, E., Roth, D.: Understanding the value of features for coreference resolution. In: Proceedings of EMNLP 2008, pp. 294–303 (2008)

Denis, P., Baldridge, J.: Specialized models and ranking for coreference resolution. In: Proceedings of EMNLP 2008, pp. 660–669 (2008)

Denis, P., Baldridge, J.: Global joint models for coreference resolution and named entity classification. Procesamiento del Lenguaje Natural 42(1), 87–96 (2009)

Hall, M., Eibe, F., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.: The WEKA Data Mining Software: An Update. ACM SIGKDD Explorations Newsletter 11(1), 10–18 (2009)

Lowd, D., Domingos, P.: Efficient weight learning for markov logic networks. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) PKDD 2007. LNCS (LNAI), vol. 4702, pp. 200–211. Springer, Heidelberg (2007)

Luo, X., Ittycheriah, A., Jing, H.: A mention-synchronous coreference resolution algorithm based on the bell tree. In: Proceedings of ACL 2004 (2004)

Martschat, S., Cai, J., Broscheit, S., Mújdricza-Maydt, E., Strube, M.: A Multigraph Model for Coreference Resolution. In: Proceedings of EMNLP-CoNLL 2012: Shared Task, pp. 100–106 (2012)

Ng, V., Cardie, C.: Improving machine learning approaches to coreference resolution. In: Proceedings of ACL 2002, pp. 104–111 (2002)

Niu, F., Ré, C., Doan, A., Shavlik, J.: Tuffy: scaling up statistical inference in Markov logic networks using an RDBMS. Proceedings of the VLDB Endowment 4(6), 373–384 (2011)

Poesio, M., Alexandrov-Kabadjov, M., Vieria, R., Goulart, R., Uryupina, O.: Does discourse-new detection help definite description resolution? In: Proceedings of IWCS, vol. 6, pp. 236–246 (2005)

Poon, H., Domingos, P.: Joint unsupervised coreference resolution with Markov logic. In: Proceedings of EMNLP 2008, pp. 650–659 (2008)

Pradhan, S., Moschitti, A., Xue, N.: CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In: Proceedings of EMNLP-CoNLL: Shared Task, pp. 1–27 (2012)

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

Richardson, M., Domingos, P.: Markov logic networks. Machine Learning 62(1), 107–136 (2006)

Ritz, J.: Using tf-idf-related Measures for Determining the Anaphoricity of Noun Phrases. In: Proceedings of KONVENS 2010, pp. 85–92 (2010)

Song, Y., Jiang, J., Zhao, X., Li, S., Wang, H.: Joint Learning for Coreference Resolution with Markov Logic. In: Proceedings of EMNLP-CoNLL 2012, pp. 1245–1254 (2012)

Soon, W., Ng, H., Li, D.: A machine learning approach to coreference resolution of noun phrases. Computational Linguistics 27(4), 521–544 (2001)

Wellner, B., McCallum, A., Peng, F., Hay, M.: An integrated, conditional model of information extraction and coreference with application to citation matching. In: Proceedings of UAI 2004, pp. 593–601 (2004)