Coreference Based Event-Argument Relation Extraction on Biomedical Text

Katsumasa Yoshikawa
NAIST, Japan
katsumasa-y@is.naist.jp

Sebastian Riedel
University of Massachusetts, Amherst
sebastian.riedel@gmail.com

Tsutomu Hirao
NTT CS Lab, Japan
hirao@cslab.kecl.ntt.co.jp

Masayuki Asahara
NAIST, Japan
masayu-a@is.naist.jp

Yuji Matsumoto
NAIST, Japan
matsu@is.naist.jp

Abstract

This paper presents a new approach that exploits coreference information to extract event-argument (E-A) relations from biomedical documents. This approach has two advantages: (1) it can extract a large number of valuable E-A relations based on the concept of *salience in discourse* (Grosz et al., 1995) ; (2) it enables us to identify E-A relations over sentence boundaries (cross-links) using *transitivity* involving coreference relations. We propose two coreference-based models: a pipeline based on Support Vector Machine (SVM) classifiers, and a joint Markov Logic Network (MLN). We show the effectiveness of these models on a biomedical event corpus. The both models outperform the systems without coreference information. When compared with the two models, joint MLN outperforms pipeline SVM with gold coreference information.

1 Introduction

The increasing amount of biomedical texts generated by high throughput experiments demands to extract useful information automatically by Natural Language Processing techniques. One of the more recent information extraction tasks is biomedical event extraction. With the introduction of the GENIA Event Corpus (Kim et al., 2008) and the BioNLP'09 shared task data (Kim et al., 2009), a set of documents annotated with events and their arguments, various approaches for event extraction have been proposed so far (Björne et al., 2009; Buyko et al., 2009; Poon and Vanderwende, 2010).

However, previous work has only considered the problem on a per-sentence basis, neglecting possible information from other sentences in the same document we may be able to exploit. In particular, no one has yet considered using coreference information to improve event extraction. Here we propose a new approach to extract event-argument (E-A) relations that does make use of coreference information.

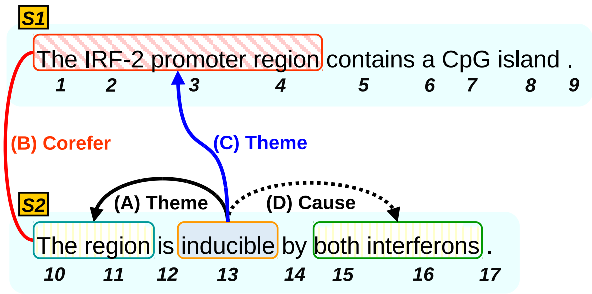

Figure 1: Cross-Sentence Event-Argument Relation Extraction

Our approach includes two main ideas:

1. aggressively extracting coreferent arguments based on *salience in discourse*
2. predicting arguments crossing sentence boundaries by *transitivity*.

First, when considering discourse structure based on Centering Theory (Grosz et al., 1995), arguments which are coreferent to something (e.g. "The region") have higher *salience in discourse*. They are hence more likely to be arguments of events mentioned in the document. Using this information helps us to identify the right arguments for candidate events and increase the likelihood of extracting arguments with antecedents corresponding to the Arrow (A) in Figure 1. Note that identifying coreferent arguments is not just important to increase F1 score on the dataset: assuming that *salience in discourse* indicates the novel information the author wants to convey, it is the set of coreferent arguments we should extract at any cost.

Secondly, previous work on this task has primarily focused on identifying event-arguments within the same sentence. See Figure 1 for an example of such cross-sentence event-argument relations. It illustrates an example of E-A relation extraction including cross-sentence E-A. In the sentence *S2*, we have "inducible" as an event to be identified. When identifying intra-sentence arguments in *S2*, we can obtain "The region" as Theme and "both interferons" as Cause. However, in this example, "The region" is not sufficient as a Theme because "The region" is coreferent to "The IRF-2 promoter region" in *S1*. Thus, the true Theme of "inducible" is "The IRF-2 promoter region" and this phrase is actually more informative as an argument. On the other hand, "The region" is just an anaphor of the true argument. *Transitivity* idea [1] allows us to extract cross-sentence E-A relations such as the Arrow (C) in Figure 1.

To implement both ideas we propose two models to extract event-argument (E-A) relations involving coreference information. One is based on local classification with SVMs, and another is based on a joint Markov Logic Network (MLNs). To remain efficient, and akin to existing approaches, both look for events on a per-sentence basis. However, in contrast to previous work, our models consider as candidate arguments not only the tokens of the current sentence, but also all tokens in the previous sentences that are identified as antecedents of some tokens in the current sentence.

We show the effectiveness of our models on a biomedical corpus. They enable us to extract cross-sentence E-A relations: we achieve an F1 score of 69.7% for our MLN model, and 54.1 % for the SVM pipeline. Moreover, with the idea of *salience in discourse* our coreference-based approach helps us to improve intra-sentence E-A extraction, in particular when arguments have antecedents. In this case adding gold coreference information to MLNs improves F-score by 16.9%.

In place of gold coreference information, we also experiment with predicted coreferences from a simple coreference resolver. Although the quality of predicted coreference information is relatively poor, we show that using this information is still better than not using it at all.

The remainder of this paper is organized as follows: Section 2 describes previous work for event extraction and some issues; Section 3 explains our proposed approach; Section 4 introduces our experimental setup; Section 5 presents results of our experiments; and in Section 6 we conclude and present some ideas for future work.

[1] e.g. If "The region" is a Theme of "inducible" and "The region" is coreferent to "The IRF-2...", then "The IRF-2..." is also a Theme of "inducible".

2 Event-Argument Relation Extraction and the Issues of Previous Work

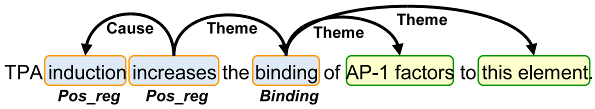

Figure 2: An Example of Biomedical Event Extraction

2.1 Biomedical Event Extraction

Event extraction on biomedical text involves the three sub-tasks; identification of event trigger words; classification of event types; extraction of the relations between events and arguments (E-A). Figure 2 shows an example of event extraction. In this example, we have three event triggers: "induction", "increases", and "binding". The corresponding event types are *Positive_regulation (Pos_reg)* for "induction" and "increases", and *Binding* for "binding". In Figure 2, "increases" has two arguments; "induction" and "binding". The roles we have to identify fall into two classes: "Theme" and "Cause". In the case of our example the roles between "increases" and the two arguments are Cause and Theme, respectively.

Note that biomedical corpora have large numbers of nominal events. For example, in Figure 2 the arguments of "increases" are both nominal events. Such events can be arguments of other events, and they are often hard to be identified.

2.2 Biomedical Corpora for Event Extraction

There are two major corpora for biomedical event extraction. One is the GENIA Event Corpus (GEC) (Kim et al., 2008), and the other is the data of the BioNLP'09 shared task. [2] This data is in fact derived from the GEC. There are some important differences between both corpora.

event type GEC has fine-grained event type annotations (35 classes), while BioNLP'09 data focuses on only 9 event subclasses.

non-event argument BioNLP'09 data does not differentiate between protein, gene and RNA, while the GEC corpus does.

[2] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/SharedTask/

coreference annotation Both GEC and BioNLP'09 corpora provide coreference annotations related to event extraction. However, in the case of the BioNLP'09 data coreference information primarily concerns protein names and abbreviations that follow in parenthesis. The GEC, on the other hand, provides proper cross-sentence coreference. Moreover, the sheer number of coreference annotations is much higher [3].

For our work we choose the GEC, primarily because of the amount and quality of coreference information it provides. This allows us to train a coreference resolver, as well as testing our hypotheses when gold coreference annotations are available. A secondary reason to prefer GEC over the BioNLP'09 corpus is its fine-grained annotation. We believe that this setting is more realistic.

2.3 Issues of Previous Work

Various approaches have been proposed for event-argument relation extraction on biomedical text. However, even the current state-of-the-art does not exploit coreference relations and focuses exclusively on intra-sentence E-A extraction.

For example, Björne et al. (2009) achieved the best results for Task 1 in the BioNLP'09 competition [4]. However, they neglected all cross-sentence E-A. They also reported that they did try to detect cross-sentence arguments directly without the use of coreference. But this approach did not lead to reasonable performance increase.

In BioNLP'09, Riedel et al. (2009) proposed a joint Markov Logic Network to tackle the task, and achieved the best results for Task 2. Their system makes use of global features and constraints, and performs event trigger and argument detection jointly. Poon and Vanderwende (2010) also applied Markov Logic and achieved competitive performance to the state-of-the-art result of (Björne et al., 2009). However, in both cases no cross-sentence information is exploited.

To summarize, so far there is no research within biomedical event extraction which exploits coreference relations and tackles cross-sentence E-A relation extraction. By contrast, for predicate-argument relation extraction in a Japanese newswire text corpus, [5] Taira et al. (2008) do consider cross-sentence E-A extraction. However, they directly extract cross-sentence links without considering coreference relations. In addition, their approach is based on a pipeline of SVM classifiers, and their achieved performance on cross-sentence E-A extraction was generally low. [6]

2.4 The Direction of Our Work

We present a new approach that exploits coreference information for E-A relation extraction. Moreover, in contrast to previous work on the BioNLP'09 shared task we apply our models in a more realistic setting. Instead of relying on gold protein annotations, we use a Named Entity tagger; and instead of focusing on the coarse-grained annotation of the BioNLP task, we work with the GEC corpus and its the fine-grained ontology.

From now on, for brevity, we call cross-sentence event-argument relations just "*cross-links*" and intra-sentence event-argument relations "*intra-links*".

We propose two coreference-based models. One is an SVM based model that extracts intra-links first and then cross-links as a post-processing step. The other is a joint model defined with Markov Logic that jointly extracts intra-links and cross-links and allows us to model salience of discourse in a principled manner.

3 Coreference Based Approach

We have two ideas for incorporating coreference information into E-A relation extraction.

- Aggressively extracting valuable E-A relations based on "*salience in discourse*"
- Predicting cross-links by using "*transitivity*" including coreference relations

According to these ideas, we propose two approaches. One is a pipeline model based on SVM classifiers, and the other is a joint model based Markov Logic.

Before we present these approaches in detail, let us first describe coreference resolution as a preprocessing step.

3.1 Coreference Resolution

There are some previous work for coreference resolution on biomedical domains (Yang et al., 2004; Su et al., 2008). However, in our work, we introduce a simple coreference resolver based on a

[3] Björne et al. (2009) also mentioned that coreference relations could be helpful to cross-sentence E-A extraction but the necessary coreference annotation to train a coreference resolver is not presented in BioNLP'09 data.

[4] BioNLP'09 has three tasks 1, 2, and 3. Task 1 is core event extraction and mandatory. Our work also focuses on Task 1.

[5] http://cl.naist.jp/nldata/corpus/

[6] Low 20s% F1

Table 1: Used Local Features for SVM Pipeline and MLN Joint

Description	SVM 1st phase *event* & *eventType*	SVM 2nd phase *role* (E-A)	MLN predicate
Word Form	X	X	$word(i, w)$
Part-of-Speech	X	X	$pos(i, p)$
Word Stem	X	X	$stem(i, s)$
Named Entity Tag	X	X	$ne(i, n)$
Chunk Tag	X	X	$chunk(i, c)$
In Event Dictionary	X	X	$dict(i, d)$
Has Capital Letter	X	X	$capital(i)$
Has Numeric Characters	X	X	$numeric(i)$
Has Punctuation Characters	X	X	$punc(i)$
Character Bigram	X		$bigram(i, bi)$
Character Trigram	X		$trigram(i, tri)$
Dependency label	X	X	$dep(i, j, d)$
Labeled dependency path between tokens		X	$path(i, j, pt)$
Unlabeled dependency path between tokens		X	$pathNL(i, j, pt)$
Least common ancester of dependency path		X	$lca(i, j, L)$

pairwise coreference model (Soon et al., 2001) [7]. It employs a binary classifier which classifies all possible pairs of noun phrases into "corefer" or "not corefer". Popular external resources like WordNet do not work in biomedical domain. Hence, our resolver identifies coreference relations with only basic features such as word form, POS, and NE tag, and achieves 59.1 pairwise F1 on GEC evaluating 5-fold cross validation.

3.2 SVM Pipeline Model

In our pipeline we apply the SVM model proposed by (Björne et al., 2009). Their original model first extracts events and event types with a multi-class SVM (1st phase). Then it identifies the relations between all pairs of event-proteins and event-events by another multi-class SVM (2nd phase). Note that, on our setting, the 1st phase classifies event types into 36 classes (35 types + "Not-Event"). Moreover, while protein annotations were given in the BioNLP'09 shared task, for the GEC we have extract them using an NE tagger. The features we used for the 1st and 2nd phases are summarized in the first and the second columns of Table 1, respectively.

After identifying intra-links, our model deterministically attaches, for each intra-sentence argument of an event, all antecedents inside/outside the sentence to the same event. Hence we implement *transitivity* as a post-processing step. However, it is difficult for SVM pipeline to implement the idea of *salience in discourse*. We believe that a Markov Logic model is preferable in this case.

3.3 MLN Joint Model

Markov Logic (Richardson and Domingos, 2006) is an expressive template language that uses weighted first-order logic formulae to instantiate Markov Networks of repetitive structure. In Markov Logic users design predicates and formulae to describe their problem. Then they use software packages such as *Alchemy* [8] and *Markov thebeast* [9] in order to perform inference and learning.

It is difficult to construct Markov Logic Networks for joint E-A relation extraction and coreference resolution across a complete document. Hence we follow the two strategies: (1) restriction of argument candidates based on coreference relations; (2) construction of a joint model which jointly identifies intra-links and cross-links. Restricting argument candidates helps us to construct a very compact but effective model. A joint model enables us to simultaneously extract intra-links and cross-links and contributes to improve the performance. In addition, we will see that this setup still allows us to implement the idea of *salience in discourse* with global formulae in Markov Logic.

3.3.1 Predicate Definition

Our joint model is based on the model proposed by Riedel et al. (2009). We first define the predicates of the proposed Markov Logic Network (MLN). There are three "*hidden*" predicates corresponding to what the target information we want to extract.

Table 2: The Three Hidden Predicates

event(i)	token i is an event
eventType(i, t)	token i is an event with type t
role(i, j, r)	token i has an argument j with role r

In this work, *role* is the primary hidden predicate because *role* represents event-argument relations.

Next we define *observed* predicates representing information that is available at both train and test

[7] Yang et al. (2004) also built the same kind of resolver as a baseline with the original coreference annotations

[8] `http://alchemy.cs.washington.edu/`

[9] `http://code.google.com/p/thebeast/`

time. We define corefer(i, j), which indicates that token i is coreferent to token j (they are in the same entity cluster). corefer(i, j) obviously plays an important role for our coreference-based joint model. We list the remaining observed predicates in the last column of Table 1.

Our MLN is composed of several weighted formulae that we divide into two classes. The first class contains local formulae for *event, eventType*, and *role*. We say that a formula is local if it considers only one atom hidden predicates. The formulae in the second class are global: they involve two or more atoms of hidden predicates. In our case they consider *event, eventType*, and *role* simultaneously.

3.3.2 Basic Local Formulae

Our local features are based on previous work (Björne et al., 2009; Riedel et al., 2009) and listed in Table 1. We exploit two types of formula representation: "simple token property" and "link tokens property" defined by Riedel et al. (2009).

The first type of local formulae describes properties of only one token and such properties are represented by the predicates in the first section of Table 1. The second type of local formulae represents properties of token pairs and linked tokens property predicates ($dep, path, pathNL$, and lca) in the second section of Table 1.

3.3.3 Basic Global Formulae

Our global formulae are designed to enforce consistency between the three *hidden* predicates and are shown in Table 3. Riedel et al. (2009) presented more global formulae for their model. However, some of these do not work well for our task setting on the GENIA Event Corpus. We obtain the best results by only using global formulae for ensuring consistency of the hidden predicates.

3.4 Using Coreference Information

We explain our coreference-based approaches with Figure 1. For our Markov Logic Network let us describe the relations in Figure 1 with predicates. First, the two intra-links in *S2* are described by role$(13, 11, \text{Theme})$ – Arrow (A) and role$(13, 15, \text{Cause})$ – Arrow (D) [10]. Next, we represent the coreference relation by corefer$(11, 4)$ – Bold Line (B). Finally, we express the cross-link as role$(13, 4, \text{Theme})$ – Arrow (C).

With the example in Figure 1, we explain the two main concepts : *Salience in Discourse (SiD)* and *Transitivity (T)*. We also present an additional idea, *Feature Copy (FC)*.

Salience in Discourse Again, an important advantage of our joint model with MLN is the implementation of "*salience in discourse*". The entities mentioned over and over again are hence important in discourse structure and accordingly it is highly possible for them to be arguments of some events.

In order to implement this idea of salience in discourse, we add the Formula (*SiD*) in the first row of Table 4. Formula (*SiD*) captures that if a token j is coreferent to another token k, there is at least one event related to token j. Our model with Formula (*SiD*) prefers coreferent arguments and aggressively connects them with events. In addition, our coreference resolver always extracts coreference relations which are related to events, since coreference annotations in GEC are always related to events.

Transitivity Another main concept is "*transitivity*" for intra/cross-link extraction. [11] As mentioned earlier, the SVM pipeline enforces *transitivity* as a post-processing step.

For the MLN joint model, let us consider the example of Figure 1 again.

$$\text{role}(13, 11, \text{Theme}) \wedge \text{corefer}(11, 4) \Rightarrow \text{role}(13, 4, \text{Theme})$$

This formula denotes that, if an event "inducible" has "The region" as a Theme and "The region" is coreferent to "The IRF-2 promoter region", then "The IRF-2 promoter region" is also a Theme of "inducible". The three atoms, role$(13, 11, \text{Theme})$, corefer$(11, 4)$, and role$(13, 4, \text{Theme})$ in this formula are respectively corresponding to the three arrow edges (A), (B), and (C) in Figure 1. This formula is generalized as Formula (*T*) shown in the second row of Table 4.

The merit of using Formula (*T*) is that we can take care of cross-links by only solving intra-links and using the associated coreference relations. Candidate arguments of cross-links are the only arguments which are coreferent to intra-sentence mentions (antecedents).

The improvement by Formula (*T*) depends on the performance of intra-link role(i, j, r) and coreference relation corefer(j, k). Clearly, this performance depends partially on the effectiveness of Formula (*T*) formula above. It should also be clear

[10] In these terms, phrasal arguments are driven by *anchor* tokens which are the ROOT tokes on dependency subtrees of the phrases

[11] An antecedent of an argument is sometimes in a subordinate clause within a same sentence

Table 3: Basic Global Formulae

Formula	Description
$event(i) \Rightarrow \exists t.eventType(i,t)$	If there is an event there should be an event type
$eventType(i,t) \Rightarrow event(i)$	If there is an event type there should be an event
$role(i,j,r) \Rightarrow event(i)$	If j plays the role r for i then i has to be an event
$event(i) \Rightarrow \exists j.role(i,j,Theme)$	Every event relates to need at least one argument.

Table 4: Coreference Formulae

Symbol	Name	Formula	Description
(*SiD*)	*Salience in Discourse*	$\mathrm{corefer}(j,k) \Rightarrow \exists i.\mathrm{role}(i,j,r) \wedge \mathrm{event}(i)$	If a token j is coreferent to another token k, there is at least one event related to token j
(*T*)	*Transitivity*	$\mathrm{role}(i,j,r) \wedge \mathrm{corefer}(j,k) \Rightarrow \mathrm{role}(i,k,r)$	If j plays the role r for i and j is coreferent to k then k also plays the role r for i
(*FC*)	*Feature Copy*	$\mathrm{corefer}(j,k) \wedge F(k,+f) \Rightarrow \mathrm{role}(i,j,r)$	If j is coreferent to k and k has feature f then j plays the role r for i

that the improvement due to Formula (*SiD*) are also affected by Formula (*T*) formula because it impacts on $\exists i.\mathrm{role}(i,j,r)$ in Formula (*SiD*). Thus, the formulae representing the *Salience in Discourse* and *Transitivity* interact with each other.

Feature Copy We implement additional usage of coreference information through "*Feature Copy*". Anaphor arguments such as "The region" in Figure 1 are sometimes more difficult to be identified than "The IRF-2 promoter region" because of the lack of basic features (e.g. POS). *Feature Copy* supplements the features of an anaphor by adding the features of its antecedent. According to the example of Figure 1, the formula,

$$\mathrm{corefer}(11,4) \wedge \mathrm{word}(4, \text{“IRF-2”}) \Rightarrow \mathrm{role}(13, 11, \mathrm{Theme})$$

injects a word feature "IRF-2" to anaphor "The region" in *S2*. Here $\mathrm{word}(i,w)$ represents a feature that the child token of the token i on the dependency subtree is word w. To be exact, this formula allows us to employ additional features of the antecedent to solve the link $\mathrm{role}(13,11,\mathrm{Theme})$. This formula is generalized as Formula (*FC*) in the last row of Table 4. In Formula (*FC*), F denotes the predicates which represent basic features such as word, POS, and NE tags of the tokens. Formula (*FC*) copies the features of cross-sentence arguments (antecedents) to intra-sentence arguments (anaphors). *Feature Copy* is not a novel idea but contributes to improve performance. The SVM pipeline model also add the same features.

4 Experimental Setup

Let us summarise the data and tools we employ. The data for our experiments is GENIA Event Corpus (GEC) (Kim et al., 2008). For feature generation, we employ the following tools. POS and NE tagging are performed with GENIA Tagger [12], for dependency path features we apply Charniak-Johnson reranking parser with Self-Training parsing model [13], and convert the results to dependency tree with pennconverter [14]. Learning and inference algorithms for joint model are provided by Markov thebeast [15], a Markov Logic engine tailored for NLP applications. Our pipeline model employs SVM-struct [16] both in learning and testing. For coreference resolution, we also employ SVM-struct for binary classification.

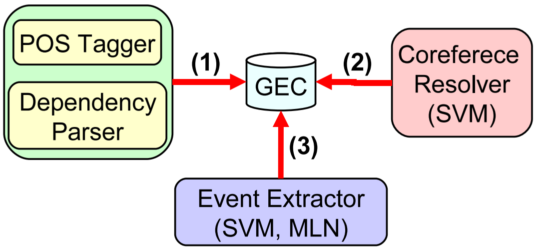

Figure 3: Figure of Experimental Setup

Figure 3 shows a structure of our experimental system. Our experiments perform the following steps. (1) First we perform preprocessing (tagging and parsing). (2) Then we perform coreference resolution for all the documents and generate lists of token pairs that are coreferent to each other. (3) finally, we train the event extractors: SVM pipeline (SVM) and MLN joint (MLN) involving coreference relations. We evaluate all systems using 5-fold cross validation on GEC.

5 Results

In the following we will first show the results of our models for event extraction with/without coreference information. We will then present more detailed results concerning E-A relation extraction.

[12] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger/

[13] http://www.cs.brown.edu/~dmcc/biomedical.html

[14] http://nlp.cs.lth.se/software/treebank_converter/

[15] http://code.google.com/p/thebeast/

[16] http://www.cs.cornell.edu/People/tj/svm_light/svm_struct.html

5.1 Impact of Coreference Based Approach

Table 5: Results of Event Extraction (F1)

System	Corefer	event	eventType	role
(a) SVM	NONE	77.0	67.8	52.3 (0.0)
(b) SVM	SYS	77.0	67.8	53.6 (**+1.3**)
(b′) SVM	GOLD	77.0	67.8	55.4 (+3.1)
(c) MLN	NONE	80.5	70.6	51.7 (0.0)
(g) MLN	SYS	80.8	70.8	53.8 (**+2.1**)
(g′) MLN	GOLD	81.2	70.8	56.7 (+5.0)

We begin by showing the SVM and MLN results for event extraction in Table 5. We present F1-values of event, eventType, and role (E-A relation). The three columns (event, eventType, and role) in Table 5 correspond to the *hidden* predicates in Table 2.

Let us consider the rows of (a)-(b) and (c)-(g). They compare SVM and MLN approaches with and without the use of coreference information. The column "Corefer" indicates how to include coreference information: "NONE"– without coreference; "SYS"– with coreference resolver; "GOLD"– with gold coreference annotations.

We note that adding coreference information leads to 1.3 points F1 improvement for the SVM pipeline, and a 2.1 points improvement for MLN joint. Both improvements are statistically significant. [17] With gold coreference information, Systems (b′) and (g′) clearly achieve more significant improvements.

Let us move on to the comparisons between SVM pipeline and MLN joint models. For event and eventType we compare row (b) with row (g) and observe that the MLN outperforms the SVM. This is to be contrasted with results for the BioNLP'09 shared task, where the SVM model (Björne et al., 2009) outperformed the MLN (Riedel et al., 2009). This contrast may stem from the fact that GEC events are more difficult to extract due to a large number of event types and lack of gold protein annotations, and hence local models are more likely to make mistakes that global consistency constraints can rule out.

For role extractions (E-A relation), SVM pipeline and MLN joint show comparable results, at least when not using coreference relations. However, when coreference information is taken into account, the MLN profits more. In fact, with gold coreference annotations, the MLN outperforms SVM pipeline by 1.3 points margin.

[17] $\rho < 0.01$, McNemar's test 2-tailed

5.2 Detailed Results for Event-Argument Relation Extraction

Table 6 shows the three types of E-A relations we evaluate in detail.

Table 6: Three Types of Event-Argument

Type	Description	Edge in Figure 1
Cross	E-A relations crossing sentence boundaries (cross-link)	Arrow (C)
W-ANT	Intra-sentence E-As (intra-link) with antecedents	Arrow (A)
Normal	Neither Cross nor W-ANT	Arrow (D)

They correspond to the arrows (A), (C), and (D) in Figure 1, respectively. We show the detailed results of E-A relation extraction in Table 7. The all scores shown in the table are F1-values.

Table 7: Results of E-A Relation Extraction (F1)

System	Corefer	Cross	W-ANT	Normal
(a) SVM	NONE	0.0	56.0	53.6
(b) SVM	SYS	**27.9**	57.0	54.3
(b′) SVM	GOLD	**54.1**	57.3	55.4
(c) MLN	NONE	0.0	49.8 (0.0)	53.2
(d) MLN	*FC*	0.0	51.5 (+1.7)	53.7
(e) MLN	*FC+SiD*	0.0	54.6 (+4.8)	53.3
(f) MLN	*FC+T*	36.7	51.7 (+1.9)	53.7
(g) MLN	*FC+SiD+T*	**39.3**	56.5 (**+6.7**)	54.3
(g′) MLN	GOLD	**69.7**	66.7 (**+16.9**)	55.3

5.2.1 SVM pipeline Model

The first part of Table 7 shows the results of the SVM pipeline with/without coreference relations. Systems (a), (b) and (b′) correspond to the first three rows in Table 5, respectively. We note that the SVM pipeline manages to extract cross-links with an F1 score of 27.9 points with coreference information from the resolver. The third row in Table 7 shows the results of the system with gold coreference which is extended from System (b). With gold coreference, the SVM pipeline achieves 54.1 points for "Cross". However, the improvement we get for "W-ANT" relations is small since the SVM pipeline model employs only *Feature Copy* and *Transitivity* concepts. In particular, it cannot directly exploit *Salience in Discourse* as a feature.

5.2.2 MLN joint Model

How does coreference help our MLN approach? To answer this question, the second part of Table 7 shows the results of the following six systems. The row (c) corresponds to the fourth row of Table 5 and shows results for the system that does not exploit any coreference information. Systems (d)-(g) include Formula (*FC*). In the sixth (e) and the seventh (f) rows, we show the scores of MLN joint with Formula (*SiD*) and Formula (*T*), respectively. Our full joint model with both (*SiD*) and (*T*) formulae comes in the eighth row (g). System (g′)

is an extended system from System (g) with gold coreference information.

By comparing Systems (d)(e)(f) with System (c), we note that *Feature Copy (FC)*, *Salience in Discourse (SiD)*, and *Transitivity (T)* formulae all successfully exploit coreference information. For "W-ANT", Systems (d) and (e) outperform System (c), which establishes that both *Feature Copy* and *Salience in Discourse* are sensible additions to an MLN E-A extractor. On the other hand, for "Cross (cross-link)", System (f) extracts cross-sentence E-A relations, which demonstrates that *Transitivity* is important, too. Next, for cross-link, our full system (g) achieved 39.3 points F1 score and outperformed System (c) with 6.7 points margin for "W-ANT". The further improvements with gold coreference are shown by our full system (g′). It achieved 69.7 points for "Cross" and improved System (c) by 16.9 points margin for "W-ANT".

5.2.3 SVM Pipeline VS MLN Joint

The final evaluation compares SVM pipeline and MLN joint models. Let us consider Table 7 again. When comparing System (a) with System (c), we notice that the SVM pipeline (a) outperforms the MLN joint model in "W-ANT" without coreference information. However, when comparing Systems (b) and (g) (using coreference information by the resolver), MLN result is very competitive for "W-ANT", 11.4 points better for "Cross".

Furthermore, with gold coreference, the MLN joint (System (g′) outperforms the SVM pipeline (System (b′)) both in "Cross" and "W-ANT" by 15.6 points margin and 9.4 points margin, respectively. This demonstrates that our MLN model will further improves extraction of cross-links and intra-links with antecedents if we have a better coreference resolver.

We believe that the reason for these results are two crucial differences between the SVM and MLN models:

- With Formula (*SiD*) in Table 4, MLN joint has more chances to extract "W-ANT" relations. It also effects the first term of Formula (*T*). By contrast, the SVM pipeline cannot easily model the notion of *salience in discourse* and the effect from coreference is weak.
- Formula (*T*) of MLN is defined as a soft constraint. Hence, other formulae may reject a suggested cross-link from Formula (*T*). The SVM pipeline deterministically identifies cross-links and is hence more prone to errors in the intra-sentence E-A extraction.

Finally, the potential for further improvement through coreference-based approach is limited by the performance on intra-links extraction. Moreover, we also observe that the 20% of cross-links are cases of zero-anaphora. Here the utility of coreference information is naturally limited, and our Formula (*T*) cannot come into effect due to missing corefer(j, k) atoms.

6 Conclusion and Future Work

In this paper we presented a novel approach to event extraction with coreference relations. Our approach incorporates coreference relations with two concepts of *salience in discourse* and *transitivity*. The coreferent arguments we focused on are generally valuable for document understanding in terms of discourse structure and they should be aggressively extracted. We proposed two models: SVM pipeline and MLN joint and they improved the attachments of intra-sentence and cross-sentence related to coreference relations. Furthermore, we confirmed that the more improvements of coreference resolution led to the higher performance of event-argument relation extraction.

However, potential for further improvement through coreference-based approach is limited by the performance of intra-sentence links and zero-anaphora cases. To overcome this problem, we plan to propose a collective approach for a whole document. Specifically, we are constructing a joint model of coreference resolution and event extraction considering all tokens in a document based on the idea of Narrative Schema (Chambers and Jurafsky, 2009). If we take into account of all tokens in a document at one time, we can consider various relations between events (event chains) through anaphoric chains. But to implement such a joint model by Markov Logic, we cannot escape from fighting against time and space complexities. So, we are investigating a reasonable approximation for learning and inference of joint approaches.

References

Jari Björne, Juho Heimonen, Filip Ginter, Antti Airola, Tapio Pahikkala, and Tapio Salakoski. 2009. Extracting complex biological events with rich graph-based feature sets. In *BioNLP '09: Proceedings of the Workshop on BioNLP*, pages 10–18, Boulder, CO, USA. Association for Computational Linguistics.

Ekaterina Buyko, Erik Faessler, Joachim Wermter, and Udo Hahn. 2009. Event extraction from trimmed dependency graphs. In *BioNLP '09: Proceedings of the Workshop on BioNLP*, pages 19–27, Boulder, CO, USA. Association for Computational Linguistics.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. 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*, pages 602–610, Suntec, Singapore, August. Association for Computational Linguistics.

Barbara J. Grosz, Scott Weinstein, and Aravind K. Joshi. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21:203–225.

Jin-Dong Kim, Tomoko Ohta, and Jun'ichi Tsujii. 2008. Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, 9(1):10+.

Jin-Dong Kim, Tomoko Ohta, Sampo Pyysalo, Yoshinobu Kano, and Jun'ichi Tsujii. 2009. Overview of bionlp'09 shared task on event extraction. In *BioNLP '09: Proceedings of the Workshop on BioNLP*, pages 1–9, Boulder, CO, USA. Association for Computational Linguistics.

Hoifung Poon and Lucy Vanderwende. 2010. Joint inference for knowledge extraction from biomedical literature. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 813–821, Los Angeles, California, June. Association for Computational Linguistics.

Matthew Richardson and Pedro Domingos. 2006. Markov logic networks. *Machine Learning*, 62(1-2):107–136.

Sebastian Riedel, Hong-Woo Chun, Toshihisa Takagi, and Jun'ichi Tsujii. 2009. A markov logic approach to bio-molecular event extraction. In *BioNLP '09: Proceedings of the Workshop on BioNLP*, pages 41–49, Boulder, CO, USA. Association for Computational Linguistics.

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.

Jian Su, Xiaofeng Yang, Huaqing Hong, Yuka Tateisi, and Jun'ichi Tsujii. 2008. Coreference resolution in biomedical texts: a machine learning approach. In Michael Ashburner, Ulf Leser, and Dietrich Rebholz-Schuhmann, editors, *Ontologies and Text Mining for Life Sciences : Current Status and Future Perspectives*, number 08131 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

Hirotoshi Taira, Sanae Fujita, and Masaaki Nagata. 2008. A japanese predicate argument structure analysis using decision lists. In *EMNLP '08: Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 523–532, Honolulu, HI, USA. Association for Computational Linguistics.

Xiaofeng Yang, Guodong Zhou, Jian Su, and Chew Lim Tan. 2004. Improving noun phrase coreference resolution by matching strings. In *Proceedings of 1st Internation Joint Conference of Natural Language Processing*, pages 326–333.