Published in: In Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) -- Long Papers. - Association for Computational Linguistics, 2013. Pp. 119-130.

Towards Weakly Supervised Resolution of Null Instantiations

Philip Gorinski
Saarland University
philipg@coli.uni-saarland.de

Josef Ruppenhofer
Hildesheim University
ruppenho@uni-hildesheim.de

Caroline Sporleder
Trier University
sporledc@uni-trier.de

Abstract

This paper addresses the task of finding antecedents for locally uninstantiated arguments. To resolve such null instantiations, we develop a weakly supervised approach that investigates and combines a number of linguistically motivated strategies that are inspired by work on semantic role labeling and corefence resolution. The performance of the system is competitive with the current state-of-the-art supervised system.

1 Introduction

There is a growing interest in developing algorithms for resolving locally unrealized semantic arguments, so-called *null instantiations (NIs)*. Null instantiations are frequent in natural discourse; only a relatively small proportion of the theoretically possible semantic arguments tend to be locally instantiated in the same clause or sentence as the target predicate. This even applies to core arguments of a predicate i.e., those that express participants which are necessarily present in the situation which the predicate evokes. However, null instantiated arguments can often be 'recovered' from the surrounding context.

Consider example (1) below (taken from Arthur Conan Doyle's "The Adventure of Wisteria Lodge"). In a frame-semantic analysis of (1), *interesting* evokes the `Mental_stimulus_stimulus_focus` (`Mssf`) frame. This frame has two core semantic arguments, EXPERIENCER and STIMULUS, as well as eight peripheral arguments, such as TIME, MANNER, DEGREE. Of the two core arguments, neither is realized in the same sentence. Only the peripheral argument DEGREE (DEG) is instantiated and realized by *most*. To fully comprehend the sentence, it is necessary to infer the fillers of the EXPERIENCER and STIMULUS roles, i.e., the reader needs to make an assumption about what is interesting and to whom. For humans this inference is easy to make since the EXPERIENCER (EXP) and STIMULUS (STIM) roles are actually filled by *he* and *a white cock* in the previous sentence. Similarly, in (2) *right* evokes the `Correctness` (`Corr`) frame, which has four core arguments, only one of which is filled locally, namely SOURCE (SRC), which is realized by *You* (and co-referent with *Mr. Holmes*). However, another argument, INFORMATION (INF), is filled by the preceding sentence (spoken by a different speaker, namely Holmes), which provides details of the fact about which Holmes was right.

(1) ["A white cock,"]$_{Stim}$ said [he]$_{Exp}$. "[Most]$_{Deg}$ **interesting**$_{Mssf}$!"

(2) A. ["Your powers seem superior to your opportunities."]$_{Inf}$
B. "[You]$_{Src}$'re **right**$_{Corr}$, Mr. Holmes."

Semantic role labeling (SRL) systems typically only label arguments that are locally realised (e.g., within the maximal projection of the target predicate); they tacitly ignore all roles that are not instantiated locally. Previous attempts to resolve null instantiated arguments have obtained mixed results. While Gerber and Chai (2010, 2012) obtain reasonable results for NI resolution within a restricted PropBank-based scenario, the accuracies obtained on the FrameNet-based data set provided for the SemEval 2010

Shared Task 10 (Ruppenhofer et al., 2010; Chen et al., 2010; Tonelli and Delmonte, 2010, 2011; Silberer and Frank, 2012) are much lower. This has two reasons: Semantic role labelling in the FrameNet framework is generally harder than in the PropBank framework, even for overt arguments, due to the fact that FrameNet roles are much more grounded in semantics as opposed to the shallower, more syntactically-driven PropBank roles. Second, the SemEval 2010 data set consists of running text in which null instantiations are marked and resolved, while the data set used by Gerber and Chai (2010, 2012) consists of annotated examples sentences for just a few predicates. This makes the latter data set easier as there are fewer predicates to deal with and more examples per predicate to learn from. However, this set-up is somewhat artificial and unrealistic (Ruppenhofer et al., to appear). Independently of whether the NI annotation is done on individual predicates or running texts, it is unlikely that we will ever have sufficient amounts of annotated data to address large-scale NI resolution in a purely supervised fashion.

In this paper, we present a system that uses only a minimal amount of supervision. It combines various basic NI resolvers that exploit different types of linguistic knowledge. Most of the basic resolvers employ heuristics; however, we make use of semantic representations of roles learnt from FrameNet. Note that the system does not require data annotated with NI information, only data annotated with overt semantic roles (i.e., FrameNet). Our paper is largely exploratory; we aim to shed light on what types of information are useful for this task. Similarly to Silberer and Frank (2012) , we focus mainly on NI *resolution*, i.e., we assume that it is known whether an argument is missing, which argument is missing, and whether the missing argument has a definite or indefinite interpretation (DNI vs. INI, see Section 2 for details).[1]

2 Arguments and Null Instantiations in FrameNet

A predicate argument structure in FrameNet consists of a *frame* evoked by a target predicate. Each frame defines a number of *frame elements* (FEs). For some FEs, FrameNet explicitly specifies a *semantic type*. For instance, the EXPERIENCER of the `Mental_stimulus_stimulus_focus` frame (see (1)) is defined to be of type 'sentient'. We make use of this information in the experiments. The FEs are categorized into core arguments, peripheral arguments, and extra-thematic arguments. *Core arguments* are taken to be essential components of a frame; they distinguish it from other frames and represent participants which are necessarily present in the situation evoked by the frame, though may not be overtly realized every time the frame is evoked. *Peripheral arguments* are optional and generalize across frames, in that they can be found in all semantically appropriate frames. Typical examples of peripheral arguments are TIME or MANNER. Finally, *extra-thematic arguments* are those that situate the event described by the target predicate against another state-of-affairs. For example, *twice* can express the extra-thematic argument ITERATION. Since only core arguments are essential to a frame, only they are analyzed as null instantiated if missing. Peripheral and extra-thematic arguments are optional by definition.

(3) [A drunk burglar]$_{Sspct}$ was **arrested**$_{Arrest}$ after accidentally handing his ID to his victim.

(4) [We]$_{Thm}$ **arrived**$_{Arrive}$ [at 8pm]$_{Tm}$.

NIs can be classified into definite NIs (DNIs) or indefinite NIs (INI). The difference is illustrated by examples (3) and (4). Whereas, in (3) the protagonist making the arrest is only existentially bound within the discourse (an instance of indefinite null instantiation, INI), the GOAL location in (4) is an entity that must be accessible to speaker and hearer from the discourse or its context (definite null instantiation, DNI). As INIs do not need to be accessible within a context, the task of resolving NIs is restricted to DNIs. The complete task can then be modeled as a pipeline consisting of three sub-tasks: (i) identifying potential NIs by taking into account information about core arguments, (ii) automatically distinguishing between DNIs and INIs, and (iii) resolving NIs classified as DNI to a suitable referent in the text. In this paper, we focus largely on the last subtask.

[1]The first two questions are the focus of recent work on motion predicates by Feizabadi and Padó (2012).

3 Related work

Null instantiations were the focus of the SemEval-10 Task-10 (Ruppenhofer et al., 2010). The two participating systems which addressed the NI resolution task took very different approaches. Tonelli and Delmonte (2010) developed a knowledge-based system called VENSES++ that builds on an existing text understanding system (Delmonte, 2008). Different resolution strategies are employed for verbal and nominal predicates. For the former, NIs are resolved by reasoning about the semantic similarity between an NI and a potential filler using WordNet. For nominal predicates, the system makes use of a common sense reasoning module that builds on ConceptNet (Liu and Singh, 2004). The system is conservative and has a relatively high precision but a low recall, identifying less than 20% of the NIs correctly. To address the low recall, Tonelli & Delmonte in later work (Tonelli and Delmonte, 2011) developed a simpler role linking strategy that is based on computing a relevancy score for the nominal head of each potential antecedent. The intuition is that heads which serve often as role fillers and occur close to the target NI are more likely to function as antecedents for the NI. Compared to the earlier model, the new method led to a noticeable increase in recall and f-score but a drop in precision.

The second SemEval system (Chen et al., 2010) is statistical and extends an existing semantic role labeler (Das et al., 2011). Resolving DNIs is modeled in the same way as labeling overt arguments, with the search space being extended to pronouns, NPs, and nouns outside the sentence.[2] When evaluating a potential filler, the syntactic features which are used in argument labeling of overt arguments are replaced by two semantic features: The system checks first whether a potential filler in the context fills the null-instantiated role overtly in one of the FrameNet sentences, i.e. whether there is a precedent for a given filler-role combination among the overt arguments of the frame in FrameNet. If not, the system calculates the distributional similarity between filler and role. The surface distance between a potential filler and an NI is also taken into account. While Chen et al.'s system has a higher recall than VENSES++, its performance is still relatively low. The authors argue that data sparseness is the biggest problem.

Silberer and Frank (2012) also used supervised machine learning to model NI resolution for the SemEval data. However, while Tonelli & Delmonte and Chen et al. view NI resolution as an extension of semantic role labelling, Silberer and Frank explicitly cast the problem as a coreference resolution (CR) task, employing an entity-mention model, i.e. the potential fillers are taken to be entity chains rather than individual mentions of discourse referents. They experiment with a variety of features, both from SRL and CR and automatically expand the training set with examples generated from a coreference corpus. They find that CR features, such as salience, perform somewhat better than SRL features.

Gerber and Chai (2010; 2012) present a study of implicit arguments for a group of frequent nominal predicates. They also use an entity mention approach and model the problem as a classical supervised task, implementing a number of syntactic, semantic, and discourse features such as the sentence distance between an NI and its potential filler, their mutual information, and the discourse relation holding between the spans containing the target predicate and the potential filler. Gerber and Chai report results that are noticeably higher than those obtained for the SemEval data. However, this is probably largely due to the fact that the two data sets are very different. Gerber and Chai's corpus consists of newswire texts (Wall Street Journal), which are annotated with NomBank/PropBank roles. The data cover 10 nominal predicates from the commerce domain, with—on average—120 annotated instances per predicate. The Task-10 corpus consists of narrative texts annotated under the FrameNet paradigm. Crucially, this corpus provides annotations for running texts not for individual occurrences of selected target predicates. It thus treats many different general-language predicates of all parts of speech. While the overall size of the corpus in terms of sentences is comparable to Gerber and Chai's corpus, the SemEval corpus contains many more target predicates and fewer instances for each.[3] NI resolution results obtained by the Task-10 participants are significantly below those reported by Gerber and Chai (2010).

[2] This disregards other role fillers such as whole sentences as in example (2) above.

[3] E.g., Ruppenhofer et al. (2010) report that there are 1,703 frame instances covering 425 distinct frame types, which gives an average of 3.8 instances per frame.

data set	sentences	tokens	frame instances	frame types	overt frame elements	DNIs (resolved)	INIs
Wisteria	438	7,941	1,370	317	2,526	303 (245)	277
Hound	525	9,131	1,703	452	3,141	349 (259)	361

Table 1: Statistics for the SemEval-10 Task-10 corpus

4 Data

In our experiments we used the corpus distributed for SemEval-10's Task-10 on "Linking Events and Their Participants in Discourse" (Ruppenhofer et al., 2010). The data set consists of two texts by Arthur Conan Doyle, "The Adventure of Wisteria Lodge"(1908) and "The Hound of the Baskervilles" (1901/02). The annotation consists of frame-semantic argument structure, co-reference chains, and information about null instantiation, i.e., the NI type (DNI vs. INI) and the filler, if available in the text. Table 1 provides basic statistics about this data set.

The Wisteria data were given out for training in the SemEval task. We use these data for parameter tuning and error analysis. We also use the overt FE annotations in Wisteria to compute semantic vectors of roles. For comparison with previous systems, the final results we report are for the unseen Hound data (the test set in SemEval).

5 Modeling NI Resolution

While the complete NI resolution task consists of three steps, detecting NIs, classifying NIs as DNIs or INIs, and resolving DNIs, in this paper, we focus exclusively on the third task as this is by far the most difficult one. We model the problem as a weakly supervised task, where the only type of supervision is the use of a corpus annotated with overtly realised semantic roles. We do not make use of the NI annotations in the training set. This distinguishes our work from the approaches by Gerber and Chai (2012; 2010) and Silberer and Frank (2012). However, like these two we employ an entity mention model, that is, we take into account the whole coreference chain for a discourse entity when assessing its likelihood of filling a null instantiated role. For this, we make use of the gold standard coreference chains in the SemEval data. So as not to have an unfair advantage, we also create singleton chains for all noun phrases without an overt co-referent, since such cases could, in theory, be antecedents for omitted arguments. Finally, since NIs can also refer to complete sentences, we augment the entity set by all sentences in the document.

We implemented four linguistically informed resolvers plus a baseline resolver. Each resolver returns the best antecedent entity chain according to its heuristics or null, if none can be found. If two or more chains score equally well for a given resolver, the one whose most recent mention is closest to the target predicate is chosen, i.e., we employ recency/salience as a tie breaker. To arrive at the final decision over the output of all (informed) resolvers, we experimented with various weighting schemes but found that majority weighting works best.

5.1 Semantic Type Based Resolver (Stres)

One approach we pursue for identifying a suitable mention/chain relies on the semantic types that FrameNet specifies for frame elements. Specifically, we look up in FrameNet the semantic type(s) of the FE that is unexpressed. With that information in hand, we consider all the coreference chains that are active in some window of context, where being active means that one of the member mentions of the chain occurs in one of the context sentences. We try to find chains that share at least one semantic type with the FE in question. This is possible because for each chain, we have percolated the semantic types

associated with any of their member mentions to the chain.[4] If we find no chain at all within the window that has semantic types compatible with our FE, we guess that the FE has no antecedent.[5] Note also that in our current set-up we have defined the semantic type match to be a strict one. For instance, if our FE has the semantic type *Entity* and an active chain is of the type *Sentient*, we will not get a match even though the type *Sentient* is a descendant of *Entity* in the hierarchy in which semantic types are arranged.

5.2 String Based Resolver (String)

Another way of finding a correct filler is the frame-independent search for realizations of the null instantiated frame element in a given context window. This is based on the assumption that a constituent which has taken a given role before is likely to fill out that role again.

An example is (5), where *house* fills the role of GOAL in an instance of the `Cotheme` frame evoked by *led* and is the correct antecedent for the omitted GOAL FE in a later instance of the `Arriving` frame.

(5) s2: The curved and shadowed drive **led**$_{Cotheme}$ us [to a low , dark house , pitchblack against a slate-coloured sky]$_{Goal}$. ... s11: "I am glad you have **come**$_{Arriving}$, sir . ..."

Investigating the active chains in the context, we try to find any chain containing a mention that is annotated with a frame element of the same name as the null instantiated FE. We do so concentrating on the FE name only and disregard the actual annotated frame, making use of the observation that FrameNet tends to assign similar names to similar roles across frames. In our current set-up, the matching of FE names is strict. Note that this constraint could be weakened by also considering frame elements that have *similar* names to the FE under investigation. For example, many 'numbered' FE names such as PROTAGONIST_1 could be treated as equivalent to simple unnumbered names such as PROTAGONIST. Note that a similar feature is used by Chen et al. (2010). The difference is that they compute the feature on the FrameNet data while we use the SemEval data.

5.3 Participant Based Resolver (Part)

Instead of concentrating on the null instantiated FE itself, another approach is to investigate the other participants of the frame in question. Based on the assumption that roles occuring together with similar other roles can be instantiated with the same filler, we search the coreference chains for mentions with the highest overlap of roles with the frame under investigation. For this, the set of roles excluding the null instantiated FE is checked against the role sets of frames in the context window. In case of an overlap between those sets, we choose the mention as a possible filler that is annotated with an FE that is not in the set. In case of there being multiple mentions fulfilling this criterion, the mention closest to the NI is chosen. The mention that is finally chosen as the filler is that mention whose annotated frame shares the most participants with the null instantiation's frame.

5.4 Vector Based Resolver (Vec)

Another semantics-based approach next to the Semantic Type Based Resolver is to calculate the similiarity between the mentions in a coreference chain and the known fillers of a null instantiated frame element. For each annotated (overt) FE in FrameNet and Wisteria, we calculate a context vector for the filler's head word, consisting of the 1000 most frequent words in the English Gigaword corpus. [6] The vectors are calculated on the Gigaword corpus and the training data in addition, and the mean vector of all vectors for a particular FE fillers' head words is calculated as the target vector for said frame element. In the actual process of resolving a given null instantiation, we investigate all coreference chains in the

[4] In the official FrameNet database, not every frame element is assigned a semantic type. We modified our copy of FrameNet so that every FE does have a semantic type by simply looking up in WordNet the path from the name of a frame element to the synsets that FrameNet uses to define semantic types.

[5] Alternatively, we could have widened the window of context in the hope of hitting upon a suitable chain.

[6] http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2003T05

context window, and calculate the mean vectors of their mentions' head words. We use the cosine for measuring the similarity of each mean vector to the null instantiated frame element's vector, and choose as an antecedent the chain that bears the highest similarity. A similar feature is employed by Chen et al. (2010) who also make use of distributional similarity.

5.4.1 Baseline Resolver (Base)

The baseline resolver is based on the intuition that the (entity chain of the) mention closest to the NI might be a good filler in the absence of more sophisticated knowledge. There are essentially two filler types: NPs and sentences. The FrameNet definition of the null instantiated FE is used to determine whether its filler's semantic type should be a *living thing* or another kind of general *physical object*, in which case we link to the closest NP, or if the element is a *Topic or Message* FE, in which case we link to the preceding sentence.

6 Experiments

We first applied all individual resolvers as well as the combination of the four informed resolvers (by majority vote) to the Wisteria data set. As Table 2 shows, the string and the participant (part) resolvers behave similarly as well as the semantic type (stres) and vector (vec) resolvers: the former two have a relatively high precision but very low recall, while the latter two obtain a higher recall and f-score. This is not surprising since string and part on the one hand and stres and vec on the other hand model very similar types of information. Moreover, the string and part resolvers suffer more from sparse data since they are based on information about argument structures seen before. The more strongly semantic resolvers stres and vec are more robust.

The combination of all resolvers by majority voting outperforms each individual resolver. However, the difference is not huge, which suggests that there is a certain amount of overlap between the resolvers, i.e. they are not disjoint. We experimented with other voting schemes besides majority voting, however none led to significant improvements. As expected, the baseline resolver performs fairly poorly.

	Prec.	Rec.	F-Score	TPs
stres	0.23	0.2	0.21	51
string	0.53	0.06	0.11	16
part	0.66	0.01	0.02	2
vec	0.21	0.18	0.19	46
all	0.26	0.24	0.25	62
base	0.07	0.02	0.03	4

Table 2: Results for the individual resolvers on Wisteria

6.1 Qualitative Analysis

To shed further light on the behaviour of the resolvers as well as on the challenges of the task we performed a detailed qualitative analysis for a run on the training data in which we use a window of 3 sentences prior to the target sentence. (The results for slightly greater windows sizes up to 5 are essentially the same.)

Performance by frame For the semantic type-based resolver and the vector resolver we looked in detail at their performance on individual frames. We did not similarly look at the other two resolvers as they only identified antecedents for relatively few DNIs, thus rendering the analysis a bit unreliable. The vector and the semantic type-based resolvers behave similarly and, for reasons of space, we focus

on the latter here. We traced the system's handling of all frame instances with a DNI-FE from start to finish, providing us with detailed information on why particular cases cannot be resolved. Table 3 shows information for those FEs that are most often omitted as DNI. The resolver setting employed is one where the resolver looks backward only for coreferent mentions of the missing referent. All mentions in a window of three sentences before the DNI are considered. For instance, the first line in Table 3 shows that the FE GOAL in the `Arriving` frame occurs 14 times overall. In 12 cases, a resolution within the text is possible. However, in only 4 cases is the correct coreference chain among the set of active candidates that the resolver considers within the 3-sentence window. None of these 4 cases were resolved successfully. By comparison, performance is much higher for the FE INITIAL_SET of the `Increment` frame, where 5 of 8 resolvable instances are correctly resolved. Note that for the same frame, performance seems much lower for the FE CLASS, which, however, is also less often resolvable than its sister FE INITIAL_SET. Likewise, the numbers for WHOLE in `Calendric_unit` suggest that for some FEs in particular frames resolution to an explicit mention within the text is rarely possible and typically results in false positives. Taken together, these facts suggest that ideally we would have resolution strategies more specifically attuned to particular frame-FE combinations.

Framename	FE	Instances	Resolvable	Active	Correct
Arriving	Goal	14	12	4	0
Increment	Initial_set	9	8	5	**1**
Increment	Class	6	2	0	0
Risky_situation	Asset	6	6	5	0
Attempt	Goal	6	5	2	0
Time_vector	Landmark_event	6	3	1	0
Observable_bodyparts	Possessor	6	6	6	2
Locative_relation	Ground	5	5	4	**1**
Social_interaction_evaluation	Judge	5	4	**2**	1
Calendric_unit	Whole	5	0	0	0
		...			
Personal_relationship	Partner_2	3	3	3	**0**

Table 3: STRES performance on training data for frequent DNI-FEs (forward- and backward-looking)

Performance by search direction When resolving a DNI we considered all entity chains with mentions in a window of 3 sentences before the target predicate. We experimented with larger window sizes but this did not lead to improved performance. We also experimented with looking at the following sentences, too. In some cases, such as example (6), looking forward is the only way to get at an antecedent within a given window size (*he-his-the black-eyed , scowling , yellow devil*).

(6) s292: They pushed her into the carriage s293: She fought her way out again . s294: I took her part , got her into a cab , and here we are . s295: I shan 't forget the **face**$_{Observable\_Bodypart}$ at the carriage window as I led her away . s296: I 'd have a short life if he had his way - the black-eyed , scowling , yellow devil . "

We may thus wonder what the effect of also looking forward might be. Table 4 shows the information for the same set of frequent DNI-FEs as Table 3 but now for the resolver setting where the resolver looks forward 3 sentences as well as backward.

Comparison of the tables suggests that looking forward does not usually give us access to chains that we wouldn't have available by only looking backward. We have only one such case–Social interaction evaluation.Judge–in our tables. Overall, among the 303 DNI cases in the data, the gold chain is within range in 143 cases when we only look back and in 156 cases when we look forward, too. (+9%) Looking forward more often results in the resolution of the right candidate (chain/mention) going wrong; e.g. Increment.Initial_set is a good example from the tables above. Overall, across all cases of DNI we have a 41.9 % drop in correct resolutions.

Framename	FE	Instances	Resolvable	Active	Correct
Arriving	Goal	14	12	4	0
Increment	Initial_set	9	8	5	**5**
Increment	Class	6	2	0	0
Risky_situation	Asset	6	6	5	0
Attempt	Goal	6	5	2	0
Time_vector	Landmark_event	6	3	1	0
Observable_bodyparts	Possessor	6	6	6	2
Locative_relation	Ground	5	5	4	**2**
Social_interaction_evaluation	Judge	5	4	**1**	1
Calendric_unit	Whole	5	0	0	0
	...				
Personal_relationship	Partner_2	3	3	3	**2**

Table 4: STRES performance on training data for frequent DNI-FEs (backward-looking only)

Number of candidate chains On average there are about 26.5 different candidate chains available for a case of DNI if the system only looks back 3 sentences. Even with various constraints in place that filter out chains, the number of viable chains is still high. Consider example 7, where an antecdent needs to be found for the missing OFFENDER. That sentence alone, not including earlier ones, mentions multiple distinct human individuals and groups. Given that the correct referent (*he*) is farthest away from the frame's target, it is not surprising that resolution did not succeed given that the system has no understanding that all other mentioned individuals and groups are among the revenge-seeking PROTAGONISTs and thus highly unlikely to also fill the role of OFFENDER.

(7) s371: Knowing that he would return there , Garcia , who is the son of the former highest dignitary in San Pedro , was waiting with two trusty companions of humble station , all three fired with the same reasons for **revenge**$_{Revenge}$.

Performance by target POS The distribution of DNI cases across targets of different parts of speech is not even, as can be seen from Table 5. Neither is the performance of our systems equal for all POS, as illustrated by Table 6. On the Wisteria data resolution performance is lowest for verbs. This is somewhat surprising because traditional SRL tends to be easier for verbal predicates than for other parts-of-speech. Similarly, in our experience, we have found performance on the two steps preceding antecedent resolution, that is, on NI detection and NI-type recognition, to usually be better on verbs (and adjectives) than on nouns. However, the difference is small and may be accidental, especially since on the test data verbs, along with adjectives, again perform better than nouns.

Adjective	Noun	Prep	Adverb	Verb	Other
48	160	2	10	79	4

Table 5: Distribution of DNI instances across targets of different POS in the training data

POS	Instances	Resolvable	Gold in CandidateSet	Correct
Adj	48	38	25	8 (16.7%)
Noun	160	133	81	26 (16.25%)
Verb	79	65	33	7 (8.9%)

Table 6: Performance of the semantic type-based resolver for major POS types in the training data

Performance on specific semantic domains While our training dataset is small, we also decided to group related frames for three important semantic domains (Motion, Communication, Cognition & Perception) that are relatively frequent in the training data. We compare the resolution performance for the frame instances covered by the different groups in Table 7. Our intuition is that there may be differences between the domains. For instance, as suggested by the example of the GOAL FE in the `Arriving` frame (discussed in 6.1 above) Source and Goal FEs in motion-related frames may be relatively difficult to resolve. However, the differences between the domains are not statistically significant on the amount of data we have: the p-value of a Fisher's exact test using the Freeman-Halton extension is 0.17537655.

Domain	Instances	Resolvable	Gold in CandidateSet	Correct
Motion	33	27	11	1 (3.0%)
Communication	19	19	13	3 (15.8%)
Cognition & Perception	15	15	10	1 (6.7%)

Table 7: Resolution performance of STRES for three well-represented domains

6.2 Quantitative Analysis

For comparison with previous work, we also report our results on the SemEval test set (Hound) for the best parameter setting (majority vote, window of 5 sentences preceding the target sentence) as obtained from the development set (Wisteria). Tables 8 and 9 give the results for the role linking task only, i.e. assuming that NIs have been identified and correctly classified as DNI or INI. Tables 10 and 11 give the results for the full NI resolution task. In the latter set-up we use heuristics to identify NIs and determine DNIs. Our system is most comparable to the model by Silberer and Frank (2012), however, the latter is supervised while our model only makes use of minimal supervision. Despite this, the best results by Silberer and Frank for the role linking task are only slightly higher than ours (0.27 F1-Score). While this is encouraging, the overall performance of all NI resolution systems proposed so far for FrameNet argument structures is, of course, still relatively low. Comparing our results for the role linking (gold) vs. the full NI resolution task (non gold) indicates that there is also still room for improvement regarding NI identification and DNI vs. INI classification. The scores drop noticably for the non-gold setting. The tables below also list the performance for different parts-of-speech of the FEE. Surprisingly adjective FEEs seem to be easiest, while nouns seem more difficult than verbs. The low result for the category 'Other' can probably be explained by the fact that this category is very infrequent.

	Verb	Noun	Adj	Other	All
Precision	0.27	0.23	0.33	0.0	0.25
Recall	0.26	0.22	0.33	0.0	0.23
F1-Score	0.27	0.22	0.33	0.0	0.24

Table 8: Results on Hound Chapter 13 (gold)

	Verb	Noun	Adj	Other	All
Precision	0.32	0.22	0.38	0.0	0.27
Recall	0.29	0.21	0.33	0.0	0.24
F1-Score	0.31	0.22	0.35	0.0	0.25

Table 9: Results on Hound Chapter 14 (gold)

	Verb	Noun	Adj	Other	All
Precision	0.23	0.14	0.23	0.0	0.17
Recall	0.13	0.12	0.25	0.0	0.13
F1-Score	0.17	0.13	0.24	0.0	0.15

Table 10: Results on Hound Chapter 13 (non gold)

	Verb	Noun	Adj	Other	All
Precision	0.18	0.08	0.1	0.0	0.12
Recall	0.16	0.08	0.22	0.0	0.12
F1-Score	0.17	0.08	0.13	0.0	0.12

Table 11: Results on Hound Chapter 14 (non gold)

7 Conclusion

In this paper, we presented a weakly supervised approach to finding the antecedents for definite null instantiations. We built four different resolvers for the task, each drawing on slightly different aspects of semantics. The semantic type-based and the vector resolver focused on the properties of potential role fillers; the participant-based filler focused on the set of co-occurring roles; and the string-based resolver represents a bet that a constituent which has filled a given role before is likely to fill the same role again. While the semantic type-based and vector resolvers proved to be more robust than the others, the best system consisted in a combination of all four resolvers. The combined system produced results competitive with the current best supervised system, despite being largely unsupervised.

A detailed performance analysis for the semantic type-based resolver on the training data confirmed some prior findings and yielded several new insights into the task. First, resolution attempts could benefit from knowledge about the particulars of frames or of semantic domains. For instance, there seem to be some omissible FEs such as WHOLE in the `Calendric_unit` frame that are almost never resolvable and which we therefore might best guess to have no antecedent. Similarly, while for some FEs in some frames (e.g. INITIAL_SET in `Increment`) a very narrow window of context is sufficient, for others such as SOURCE, PATH or GOAL FEs in motion-related frames it might make sense to widen the window of context that is searched for antecedents. Second, while it is clear that definite null instantiations normally have to have prior mentions at the point when they occur, it was not obvious that also considering active chains in a window *following* the occurrence of the FEE would in fact lower performance as it does for STRES. Third, while verbs unexpectedly performed worse than nouns and adjectives on the training data, the usual pattern was observed on the test data: role labeling and NI resolution perform better on verbs than on nouns. Finally, the detailed analysis illustrates that the antecedent-finding step is indeed a hard one given that on average the correct chain has to be found among more than 25 candidates.

Acknowledgements

This work was partly funded by the German Research Foundation, DFG (Cluster of Excellence *Multimodal Computing and Interaction*).

References

Chen, D., N. Schneider, D. Das, and N. A. Smith (2010, July). SEMAFOR: Frame argument resolution with log-linear models. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, Uppsala, Sweden, pp. 264–267. Association for Computational Linguistics.

Das, D., N. Schneider, D. Chen, and N. Smith (2011). Probabilistic frame-semantic parsing. In *Proc. of NAACL-HLT-10*.

Delmonte, R. (2008). *Computational Linguistic Text Processing Lexicon, Grammar, Parsing and Anaphora Resolution*. New York: Nova Science.

Feizabadi, P. and S. Padó (2012). Automatic identification of motion verbs in wordnet and framenet. In *Proceedings of KONVENS 2012*, Vienna, Austria.

Gerber, M. and J. Y. Chai (2010). Beyond nombank: a study of implicit arguments for nominal predicates. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, Stroudsburg, PA, USA, pp. 1583–1592. Association for Computational Linguistics.

Gerber, M. and J. Y. Chai (2012). Semantic role labeling of implicit arguments for nominal predicates. *Computational Linguistics 38*(4), 755–798.

Liu, H. and P. Singh (2004). ConceptNet: A practical commonsense reasoning toolkit. *BT Technology Journal 22*(4), 211–226.

Ruppenhofer, J., R. Lee-Goldman, C. Sporleder, and R. Morante (to appear). Beyond sentence-level semantic role labeling: linking argument structures in discourse. *Language Resources and Evaluation*.

Ruppenhofer, J., C. Sporleder, R. Morante, C. Baker, and M. Palmer (2010). SemEval-2010 task 10: Linking events and their participants in discourse. In *The ACL Workshop SemEval-2010: 5th International Workshop on Semantic Evaluations*.

Silberer, C. and A. Frank (2012, 7-8 June). Casting implicit role linking as an anaphora resolution task. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, Montréal, Canada, pp. 1–10. Association for Computational Linguistics.

Tonelli, S. and R. Delmonte (2010, July). Venses++: Adapting a deep semantic processing system to the identification of null instantiations. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, Uppsala, Sweden, pp. 296–299. Association for Computational Linguistics.

Tonelli, S. and R. Delmonte (2011, June). Desperately seeking implicit arguments in text. In *Proceedings of the ACL 2011 Workshop on Relational Models of Semantics*, Portland, Oregon, USA, pp. 54–62. Association for Computational Linguistics.