PriMeSRL-Eval: A Practical Quality Metric for Semantic Role Labeling Systems Evaluation

Ishan Jindal[a], Alexandre Rademaker[a], Khoi-Nguyen Tran[a], Huaiyu Zhu[a], Hiroshi Kanayama[a], Marina Danilevsky[a], Yunyao Li[b][*]
[a]IBM Research, [b]Apple
ishan.jindal@ibm.com, alexrad@br.ibm.com, kndtran@ibm.com,
huaiyu@us.ibm.com, hkana@jp.ibm.com, mdanile@us.ibm.com,
yunyaoli@apple.com

Abstract

Semantic role labeling (SRL) identifies predicate-argument structures in a sentence. This task is usually accomplished in four steps: predicate identification, predicate sense disambiguation, argument identification, and argument classification. Errors introduced at one step propagate to later steps. Unfortunately, the existing SRL evaluation scripts do not consider the full effect of this error propagation aspect. They either evaluate arguments independent of predicate sense (CoNLL09) or do not evaluate predicate sense at all (CoNLL05), yielding an inaccurate SRL model performance on the argument classification task. In this paper, we address key practical issues with existing evaluation scripts and propose a more strict SRL evaluation metric, *PriMeSRL*. We observe that by employing PriMeSRL, the quality evaluation of all SoTA SRL models drops significantly, and their relative rankings also change. We also show that PriMeSRL successfully penalizes actual failures in SoTA SRL models.

1 Introduction

Semantic Role Labeling (SRL) extracts predicate-argument structures from a sentence, where predicates represent relations (verbs, adjectives, or nouns) and arguments are the spans attached to the predicate demonstrating "who did what to whom, when, where, and how." As one of the fundamental natural language processing (NLP) tasks, SRL has been shown to help a wide range of NLP downstream applications such as natural language inference (Zhang et al., 2020b; Liu et al., 2022), question answering (Maqsud et al., 2014; Yih et al., 2016; Zhang et al., 2020b; Dryjański et al., 2022), machine translation (Shi et al., 2016; Rapp, 2022), content moderation and verification (Calvo Figueras et al., 2022; Fharook et al., 2022), information extraction (Niklaus et al., 2018; Zhang et al., 2020a). In all of these applications, the quality of the underlying SRL models has a significant impact on the downstream tasks. Despite this, few studies exist on how to properly evaluate the quality of SRL systems in practice.

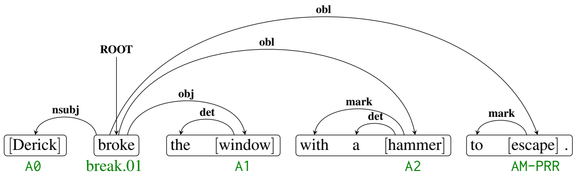

Figure 1: An SRL example with head-based semantic roles on top of Universal Dependencies annotation.

Given a sentence, a typical SRL system obtains predicate-argument structure by following a series of four steps: 1) predicate identification; 2) predicate sense disambiguation; 3) argument identification; and 4) argument classification. The predicate senses and their argument labels are taken from inventories of frame definitions such as Proposition Bank (PropBank) (Palmer et al., 2005), FrameNet (Baker et al., 1998), and VerbNet (Schuler, 2005).

The accuracy of SRL extraction is affected by the correctness of each of these steps. Consider the example in Figure 1 using PropBank[1] annotations: The SRL system must:

1. Identify the verb 'break' as a predicate

2. Disambiguate its particular sense as 'break.01', [2] which has four associated arguments: `A0` (the breaker), `A1` (thing broken), `A2` (the instrument), `A3` (the number of pieces), and `A4` (from what `A1` is broken away).[3]

[*] Work done while at IBM Research

[1] In this paper we discuss SRL based on PropBank frames.

[2] https://verbs.colorado.edu/propbank/framesets-english-aliases/break.html

[3] Note that in PropBank each verb sense has a specific set

Findings of the Association for Computational Linguistics: EACL 2023, pages 1806–1818
May 2-6, 2023 ©2023 Association for Computational Linguistics

3. Identify each argument as it occurs ('Derick', 'the window', etc.)

4. Classify the arguments ('Derick' : A0)

Finally, this example has one additional modifier: the AM-PRP (the purpose). Figure 1 illustrates the same analysis on top of the universal dependencies annotations where only the tokens' head of phrases are annotated with the proper argument.

To obtain a completely correct predicate-argument structure both the predicate sense and all of its associated arguments need to be correctly extracted. Mistakes introduced at one step may propagate to later steps, leading to further errors.

For instance, in the above example, a wrong predicate sense 'break.02' (*break in or gain entry*) has not only a different meaning from 'break.01' (*break*) but also a different set of arguments. In many cases, even if an argument for a wrong predicate sense is labeled with the same numerical roles (A1, A2, etc), their meanings can be very different. Therefore, in general, the labels for argument roles should be considered to be incorrect when the predicate sense itself is incorrect. However, existing SRL evaluation metrics (e.g. (Hajič et al., 2009)) do not penalize argument labels in such cases.

The currently used evaluation metrics also do not evaluate *discontinuous arguments* accurately. Some arguments in the PropBank original corpora have discontinuous spans that all refer to the same argument. This can happen for a number of reasons such as in verb-particle constructions. In a dependency-based analysis, these arguments end up being attached to distinct syntactic heads (Surdeanu et al., 2008). Take as an example the sentence, "I know your answer will be that those people should be allowed to live where they please as long as they pay their full locational costs." For the predicate "allow.01," the A1 (action allowed) is the discontinuous span "those people" (A1) and "to live where they please as long as they pay their full locational costs" (C-A1). Existing evaluation metrics treat these as two independent labels.

A similar problem exists for the evaluation of reference arguments (R-X). For example, in the sentence "This is exactly a road that leads nowhere", for the predicate "lead.01", the A0 "road" is referenced by C-A0 "that". If A0 is not correctly identified, the reference C-A0 is meaningless.

In this paper, we conduct a systematic analysis of the pros and cons of different evaluation metrics for SRL, including:

- Proper evaluation of predicate sense disambiguation task;
- Argument label evaluation in conjunction with predicate sense;
- Proper evaluation for discontinuous arguments and reference arguments; and
- Unified evaluation of argument head and span.

We then propose a new metric for evaluating SRL systems in a more accurate and intuitive manner in Section 3, and compare it with currently used methods in Section 4. PriMeSRL is available at https://github.com/UniversalPropositions/PriMeSRL-Eval.

2 Existing Evaluation Metrics for SRL

Most of the existing evaluation metrics came from shared tasks for the development of systems capable of extracting predicates and arguments from natural language sentences. In this section, we summarize the approaches to SRL evaluation in the shared tasks from SemEval and CoNLL.

2.1 Senseval and SemEval

SemEval (Semantic Evaluation) is a series of evaluations of computational semantic analysis systems that evolved from the Senseval (word sense evaluation) series.

SENSEVAL-3 (Litkowski, 2004) addressed the task of automatic labeling of semantic roles and was designed to encourage research into and use of the FrameNet dataset. The system would receive as input a target word and its frame, and was required to identify and label the frame elements (arguments). The evaluation metric counted the number of arguments correctly identified (complete match of span) and labeled, but did not penalize those spuriously identified. An overlap score was generated as the average of proportion of partial matches.

SemEval-2007 contained three tasks that evaluate SRL. Task 17 and 18 identified arguments for given predicates using two different role label sets: PropBank and VerbNet (Pradhan et al., 2007). They used the srl-eval.pl script from the CoNLL-2005 scoring package (Carreras and Màrquez, 2005) (see below). Task 19 consists of

of underspecified roles, given by numbers: A0, A1, A2, and so on. This is because of the well-known difficulty of defining a universal set of thematic roles (Jurafsky and Martin, 2021).

recognizing words and phrases that evoke semantic frames from FrameNet and their semantic dependents, which are usually, but not always, their syntactic dependents. The evaluation measured precision and recall for frames and frame elements, with partial credit for incorrect but closely related frames. Two types of evaluation were carried out. The first is the label matching evaluation. The participant's labeled data were compared directly with the gold standard labeled using the same evaluation procedure used in the previous SRL tasks at SemEval. The second is the semantic dependency evaluation, in which both the gold standard and the submitted data were first converted to semantic dependency graphs and compared.

SemEval-2012 (Kordjamshidi et al., 2012) and **SemEval-2013** (Kolomiyets et al., 2013) introduced the 'Spatial Role Labeling' task, but this is somewhat different from the standard SRL task and will not be discussed in this paper. Since **SemEval-2014** (Marelli et al., 2014), a deeper semantic representation of sentences in a single graph-based structure via semantic parsing has superseded the previous 'shallow' SRL tasks.

2.2 CoNLL

The **CoNLL-2004** shared task (Carreras and Màrquez, 2004) was based on the PropBank corpus, comprising six sections of the Wall Street Journal part of the Penn Treebank (Kingsbury and Palmer, 2002) enriched with predicate–argument structures. The task was to identify and label the arguments of each marked verb. The precision, recall, and F1 of arguments were evaluated using the `srl-eval.pl` program. For an argument to be correctly recognized, the words spanning the argument as well as its semantic role have to be correct. The verb argument is the lexicalization of the predicate of the proposition. Most of the time, the verb corresponds to the target verb of the proposition, which is provided as input, and only in a few cases the verb participant spans more words than the target verb. This situation makes the verb easy to identify and, since there is one verb with each proposition, evaluating its recognition overestimates the overall performance of a system. For this reason, the verb argument is excluded from evaluation. The shared task proceedings do not detail how non-continuous arguments are evaluated. In **CoNLL-2005** (Carreras and Màrquez, 2005) a system had to recognize and label the arguments of each target verb. The evaluation method remained the same as CoNLL-2004, using the same evaluation code.

The **CoNLL 2008** shared task (Surdeanu et al., 2008) was dedicated to the joint parsing of syntactic and semantic dependencies. The shared task was divided into three subtasks: (i) parsing of syntactic dependencies, (ii) identification and disambiguation of semantic predicates, and (iii) identification of arguments and assignment of semantic roles for each predicate. SRL was performed and evaluated using a dependency-based representation for both syntactic and semantic dependencies.

The official evaluation measures consist of three different scores: (i) syntactic dependencies are scored using the labeled attachment score (LAS), (ii) semantic dependencies are evaluated using a labeled F1 score, and (iii) the overall task is scored with a macro average of the two previous scores. The semantic propositions are evaluated by converting them to semantic dependencies, i.e., a semantic dependency from every predicate to all its individual arguments were created. These dependencies are labeled with the labels of the corresponding arguments. Additionally, a semantic dependency from each predicate to a virtual ROOT node was created. The latter dependencies are labeled with the predicate senses. This approach guarantees that the semantic dependency structure conceptually forms a single-rooted, connected (not necessarily acyclic) graph. More importantly, this scoring strategy implies that if a system assigns the incorrect predicate sense, it still receives some points for the arguments correctly assigned. Several additional evaluation measures were applied to further analyze the performance of the participating systems. The *Exact Match* reports the percentage of sentences that are completely correct, i.e., all the generated syntactic dependencies are correct and all the semantic propositions are present and correct. The *Perfect Proposition F1* score entire semantic frames or propositions. The ratio between labeled F1 score for semantic dependencies and the LAS for syntactic dependencies.

As in CoNLL-2008, the CoNLL-2009 shared task (Hajič et al., 2009) combined syntactic dependency parsing and the task of identifying and labeling semantic arguments of verbs or nouns for six more languages in addition to the original English from CoNLL-2008. Predicate disambiguation was still part of the task, whereas the identification of argument-bearing words was not. This decision was made to compensate for the significant differences between languages and between the annotation schemes used. The evaluation of SRL was done similar to CoNLL-2008.

3 The Proposed Approach

We propose PriMeSRL, a new metric for evaluating SRL systems, based on the following high-level rules that aim to overcome the drawbacks in existing metrics:

1. Predicate senses are considered correct only when the full predicate.sense is correct, not just the sense number. (Table 1)
2. Core arguments are considered correct only when the predicate sense has been correctly identified. (Table 1)
3. An argument of the form `C-X` is considered together with its associated `X` argument to cover the full region of the argument. (Table 2)
4. An argument of the form `R-X` is considered as a reference, so its correctness depends on the correctness of the referenced `X`. (Table 3)

3.1 Predicate sense disambiguation evaluation

Current evaluation metrics either do not evaluate the predicate sense disambiguation task (e.g. CoNLL05), or evaluate only the sense number of the predicate (e.g. CoNLL09). In this section, we only contrast with the CoNLL09 evaluation script.

To begin with, what is the predicate sense number? Similar to word senses in Wordnet (Fellbaum, 2010), the predicate senses in PropBank inside a predicate frame file are generally ordered from most to least frequently used, with the most common sense numbered 01 (Pradhan et al., 2022). The sense numbers (01, 02, 03, ...) do not have any associated semantic meaning, and merely convey that one particular meaning of the predicate is more common than another.

Therefore, predicting and evaluating only the sense number is not sensible. It can be a reasonable goal to a certain extent, such as when predicate location is given and the task is only to disambiguate the sense of the predicate (as proposed in the CoNLL09 shared task.) But the consequence of this approach is that a sense number classifier could predict a sense number that does not even exist in the associated frame file. Of course, an unknown sense number for a predicate does not have a semantic meaning, making it unsuitable for practical use cases. For a practical end-to-end SRL system, the sense number classifier should predict both the predicate location and associated sense number together (i.e. predicate.sense) so that the contextual meaning of the predicate is correctly captured, as performed in (Roth and Lapata, 2016; Li et al., 2018; Conia et al., 2021; Conia and Navigli, 2022).

# text = Yesterday, John bought a car.

ID	FORM	FLAG	PRED SENSE	Predicate-argument prediction			
				Gold	P1	P2	P3
1	Yesterday	_	_	TMP	TMP	TMP	TMP
2	,	_	_	_	_	_	_
3	John	_	_	A0	A0	A0	A0
4	bought	Y	buy.01	buy.01	*buy_out.03*	*buy.05*	*sell.01*
5	a	_	_	_	_	_	_
6	car	_	_	A1	A1	A1	A1
7	.	_	_	_	_	_	_

Predicate Evaluation	R	CoNLL05	do not evaluate			
		CoNLL09	1/1	0/1	0/1	1/1
		PriMeSRL	1/1	0/1	0/1	0/1
	P	CoNLL05	do not evaluate			
		CoNLL09	1/1	0/1	0/1	1/1
		PriMeSRL	1/1	0/1	0/1	0/1

Argument Evaluation	R	CoNLL05	3/3	3/3	3/3	0/3
		CoNLL09	3/3	3/3	3/3	3/3
		PriMeSRL	3/3	1/3	1/3	1/3
	P	CoNLL05	3/3	3/3	3/3	0/3
		CoNLL09	3/3	3/3	3/3	3/3
		PriMeSRL	3/3	1/3	1/3	1/3

Table 1: Comparing evaluation metrics on 4 examples, showing the effect of wrong predicate sense on argument label evaluation. *RED-italic* shows a wrong prediction by a hypothetical model. GREEN cell highlights where PriMeSRL differs from existing metrics.

Evaluating the predicate sense disambiguation task of such practical systems using existing evaluation metrics is not optimal. Consider the example in Figure 1, where the gold predicate.sense is 'break.01'(*break, cause to not be whole* [4]). Suppose an SRL system predicts[5] the predicate.sense label 'pull.01'(*causing motion* [6]). The existing CoNLL09 evaluation script will give a fully correct score because the predicted sense number **01** exactly matches the gold sense number, despite the different semantic meanings. In contrast, PriMeSRL evaluates the predicate.sense as a whole instead of only the sense number.

[4] https://verbs.colorado.edu/propbank/framesets-english-aliases/break.html

[5] See Table 6 for several examples of such mistakes actually made by a SoTA SRL system on CoNLL09 data.

[6] https://verbs.colorado.edu/propbank/framesets-english-aliases/pull.html

# text = Many confusing questions have been taxing my mind for years about Egypt and its people .

ID	FORM	F	PRED SENSE	Gold	P1	P2	P3	P4	P5	P6	P7
1	Many	_	_	_	_	_	_	_	_	_	_
2	confusing	_	_	_	_	_	_	_	_	_	_
3	questions	_	_	A0	A0	A1	A1	C-A0	A1	C-A0	C-A0
4	have	Y	_	_	_	_	_	_	_	_	_
5	been	Y	_	_	_	_	_	_	_	_	
6	taxing	Y	tax.01	_	_	_	_	_	_	_	_
7	my	_	_	_	_	_	_	_	_	_	_
8	mind	_	_	A2	A2	A2	A2	A2	A2	A2	A2
9	for	_	_	_	_	_	_	_	_	_	_
10	years	_	_	TMP	TMP	TMP	TMP	TMP	TMP	TMP	TMP
11	about	_	_	_	_	_	_	_	_	_	_
12	Egypt	_	_	C-A0	C-A1	C-A0	C-A1	A0	C-A2	C-A0	_
13	and	_	_	_	_	_	_	_	_	_	_
14	its	_	_	_	_	_	_	_	_	_	_
15	people	_	_	_	_	_	_	_	_	_	_
16	.	_	_	_	_	_	_	_	_	_	_

Argument HEAD Evaluation	R	Conll09	4/4	3/4	3/4	2/4	2/4	2/4	3/4	2/4
		PriMeSRL	3/3	2/3	2/3	2/3	3/3	1/3	3/3	2/3
	P	Conll09	4/4	3/4	3/4	2/4	2/4	2/4	3/4	2/3
		PriMeSRL	3/3	2/4	2/4	2/3	3/3	1/3	3/3	2/3

Argument SPAN Evaluation	R	Conll05	3/3	2/3	2/3	2/3	2/3	1/3	2/3	2/3
		PriMeSRL	3/3	2/3	2/3	2/3	3/3	1/3	3/3	2/3
	P	Conll05	3/3	2/4	2/4	2/3	2/4	1/3	2/3	2/3
		PriMeSRL	3/3	2/4	2/4	2/3	3/3	1/3	3/3	2/3

Table 2: Comparing evaluation metrics on 7 examples, showing the effect of C-X labels. *RED-italic* and GREEN cell are used in the same manner as Table 1.

3.2 Argument evaluation with incorrect predicate sense

Current metrics evaluate the arguments independent of the predicate sense. That is, they evaluate arguments as if the predicate location and sense are both correct. In practice, the predicates predicted by models can of course be wrong, and in such cases, the corresponding core argument labels (A0, A1, etc.) generally do not refer to the correct argument - even if the label itself matches the gold label - and should be penalized. Contextual arguments, or adjunct arguments, such as AM-LOC, AM-TMP, etc, remain the same across different predicates and do not need to be penalized for predicate errors.

Table 1 illustrates the difference between PriMeSRL and existing evaluation metrics, CoNLL09 and CoNLL05. For predicate sense evaluation, PriMeSRL is often equal to CoNLL09 (CoNLL05 does not measure this aspect.) PriMeSRL explicitly penalizes the cases where the lemma is wrongly identified (Example P3): The CoNLL09 script considers the label as correct as long as the "predicate sense number" is correct. It is unlikely for a model to predict "sell.01" in P3 for the gold predicate "buy.01". We choose this example to smoothly motivate the need for more strict evaluation metrics for SRL. In fact, SoTA SRL systems made this type of error. Table 6 provides some incorrect model predictions from a SoTA SRL model, where a model confuses "overheat" with "soothe" as an example.

# text = This is exactly a road that leads nowhere.

ID	FORM	F	PRED SENSE	Gold	P1	P2	P3	P4	P5	P6
1	This	_	_	_	_	_	_	_	_	_
2	is	_	_	_	_	_	_	_	_	_
3	exactly	_	_	_	_	_	_	_	_	_
4	a	_	_	_	_	_	_	A0	_	_
5	road	_	_	A0	A1	A0	A1	_	R-A0	R-A0
6	that	_	_	R-A0	R-A0	R-A1	R-A1	R-A0	R-A0	A0
7	leads	Y	lead.01	_	_	_	_	_	_	_
8	nowhere	_	_	A4	A4	A4	A4	A4	A4	A4
9	.	_	_	_	_	_	_	_	_	_

Argument HEAD Evaluation	R	Conll09	3/3	2/3	2/3	1/3	2/3	2/3	1/3
		PriMeSRL	3/3	1/3	2/3	1/3	1/3	1/3	1/3
	P	Conll09	3/3	2/3	2/3	1/3	2/3	2/3	1/3
		PriMeSRL	3/3	1/3	2/3	1/3	1/3	1/3	1/3

Argument SPAN Evaluation	R	Conll05	3/3	2/3	2/3	1/3	2/3	1/3	1/3
		PriMeSRL	3/3	1/3	2/3	1/3	1/3	1/3	1/3
	P	Conll05	3/3	2/3	2/3	1/3	2/3	1/3	1/3
		PriMeSRL	3/3	1/3	2/3	1/3	1/3	1/3	1/3

Table 3: Comparing evaluation metrics on 6 examples, showing the effect of R-X labels. *RED-italic* and GREEN cell are used in the same manner as Table 1.

For argument evaluation, both CoNLL09 head evaluation and CoNLL05 span evaluation wrongly mark all the arguments in examples P1, P2, and P3 as correct, despite the predicate sense being wrong. This is corrected by PriMeSRL.

3.3 Evaluation of C-X arguments

An argument label with prefix C- is used in situations where an argument consists of multiple non-adjacent parts (Surdeanu et al., 2008). If conceptually the whole argument should be labeled X, then operationally one part will get label X and the other parts get label C-X. The existing evaluation metrics treat all these labels as independent, which is incorrect as it increases the weight of these arguments and assigns partial credit when an exact match is required. We now describe PriMeSRL for span-based and head-based evaluations.

Span-based evaluation: For an argument split into multipart spans with labels X and C-X, the complete span can be represented by the set of all tokens identified by these labels. The full set of tokens produced by the model should be compared to the set in the gold data, and a single credit should be assigned if these sets are equal.

Head-based evaluation: An argument with X and C-X parts has these as separate heads. A model prediction is considered correct if and only if all heads for this argument are correct, in which case it is given one whole credit. This evaluation does not distinguish between X and C-X and will penalize an argument if it has extra or missing parts.

Table 2 compares PriMeSRL with ConNLL05

Model	Evaluation script	In-domain					Out-of-domain				
		PSD	Argument Classification				PSD	Argument Classification			
		F1	P	R	F1	(r)	F1	P	R	F1	(r)
(Conia et al., 2021)	CoNLL09	96.9	89.5	89.5	89.5	(3)	87.8	82.0	81.9	81.9	(3)
	PriMeSRL	95.5(↓1.4)	86.6	86.6	86.6(↓2.9)	(2)	80.9(↓6.9)	72.4	72.6	72.5(↓9.4)	(4)
(Blloshmi et al., 2021)$_{\text{nested}}$	CoNLL09	97.1	89.3	81.9	85.4	(4)	89.7	82.8	75.7	79.1	(4)
	PriMeSRL	96.4(↓0.7)	86.8	79.8	83.1(↓2.3)	(4)	86.7(↓3.0)	75.7	69.9	72.7(↓6.4)	(3)
(Blloshmi et al., 2021)$_{\text{flat}}$	CoNLL09	97.4	90.9	89.6	**90.2**	(1)	90.1	83.9	82.1	83.0	(2)
	PriMeSRL	96.9(↓0.5)	88.6	87.4	**88.0**(↓2.2)	(1)	87.8(↓2.3)	77.6	76.3	**76.9**(↓6.1)	(1)
(Jindal et al., 2022)	CoNLL09	96.8	89.9	89.3	89.6	(2)	89.8	82.9	83.1	**83.02**	(1)
	PriMeSRL	95.5(↓1.3)	86.8	86.3	86.55(↓3.0)	(3)	83.4(↓6.4)	73.9	74.3	74.1(↓8.9)	(2)

Table 4: Comparison of SoTA SRL models with PriMeSRL and CoNLL09 evaluation metrics on CoNLL09 dataset. (r) denotes the ranking of SRL models corresponding to the evaluation metric. **BOLD** shows the best model with CoNLL09 evaluation script and **BOLD** shows the best SRL model with PriMeSRL.

Dataset	Args	Train	Dev	Test	ood
CoNLL09	`C-X`	0.77	1.05	0.88	1.15
	`R-X`	1.98	2.03	2.07	2.24
CoNLL05	`C-X`	1.22	1.24	1.71	0.91
	`R-X`	3.26	3.36	3.38	2.91

Table 5: Representation of `C-X` and `R-X` arguments in each split of different SRL datasets.

and ConNLL09 on seven examples. For span evaluation, the variances among labels with and without `C-` do not penalize the result, as long as the whole span is correct. That is our proposal for counting continuation arguments is the same as the CoNLL05 evaluation script, which provides one full credit if all heads for continuation arguments are identified and labeled correctly. We only differ that we do not distinguish between `A0` and `C-A0` labels. In this manner, we are not as strict as the CoNLL05 script. For head evaluation, note that the denominators reflect the number of arguments rather than the number of split parts, and numerators count correct whole arguments.

3.4 Evaluation of `R-X` arguments

An argument label with prefix `R-` indicates a reference argument; thus, `R-X` is a reference to the argument `X`. For `R-X` to be correct, `X` must also be correct, but apart from this requirement, PriMeSRL treats them as separate arguments.

Table 3 compares evaluating `R-X` arguments using PriMeSRL with the metric used in CoNLL09 on 6 examples P1 through P6. For P1 in Table 3 (Head Evaluation), CoNLL09 gives credit for correctly identified `R-A0` for which no/incorrect `A0` is predicted, which is meaningless. The same is true for the Span evaluation script CoNLL05. However, we do not penalize the correctly labeled main argument for incorrect `R-X`.

4 Comparisons with Existing Metrics

In this section, we discuss the effectiveness of existing SRL evaluation metrics and demonstrate how PriMeSRL differs in various use cases, using SoTA neural SRL models as test models.

4.1 General settings

For simplicity of comparison with existing results, we assume the gold predicate location is given for all the experiments following Shi and Lin (2019); Jindal et al. (2020); Conia and Navigli (2022). However, PriMeSRL is able to handle missing or spurious predicates. We use Conia et al. (2021); Blloshmi et al. (2021); Jindal et al. (2022) as SoTA SRL models.

4.2 Datasets

We show the impact of evaluating with PriMeSRL on the CoNLL09 and CoNLL05 datasets. Table 5 shows the percentage of `C-X` and `R-X` arguments in each split of the different datasets. Note that these arguments make up $< 3\%$ of the total arguments; 5.09% total of the arguments in CoNLL05 test, and 2.95% in CoNLL09 test. Therefore, we expect to observe an F1 drop of at most about 3 and 5 points on the argument classification subtask due to mishandling `C-X` and `R-X` arguments for CoNLL09 and CoNLL05 datasets, respectively.

Id	Sentence	Gold	Predicted
1	He was able, now, to sit for hours in a chair in the living room and stare out at the bleak yard without moving.	stare.01	look.01
2	She greeted her husband's colleagues with smiling politeness , offering nothing.	politeness.01	minimalism.01
3	It was a Negro section of peeling row houses, store-front churches and ragged children.	peel.01	peer.01
4	He was calm, drugged , and lazy.	drug.01	dropper.01
5	The walk and his fears had served to overheat him and his sweaty armpits cooled at the touch of the night air.	overheat.01	soothe.01
6	He did not resent their supervision or Virginia's sometimes tiring sympathy.	tire.01	hiring.01

Table 6: Conia et al. (2021) model predictions on examples from CoNLL09 OOD set. All of these predicate senses are marked correct by the CoNLL09 evaluation script. PriMeSRL correctly penalizes all of these senses.

Model	Evaluation script	In-domain					Out-of-domain				
		PSD	Argument Classification				PSD	Argument Classification			
		F1	P	R	F1	(r)	F1	P	R	F1	(r)
(Zhang et al., 2021)$_{crf}$	CoNLL05	100	86.5	88.3	87.4	(3)	100	79.0	81.1	80.0	(3)
	PriMeSRL	100	86.1	87.8	87.03(↓0.4)	(2)	100	78.7	80.8	79.7(↓0.3)	(2)
(Zhang et al., 2021)$_{crf2o}$	CoNLL05	100	86.9	88.6	87.7	(2)	100	78.9	81.2	80.03	(2)
	PriMeSRL	100	86.5	88.1	**87.3**(↓0.4)	(1)	100	78.5	80.8	79.6(↓0.4)	(3)
(Jindal et al., 2022)	CoNLL05	100	87.4	88.0	**87.74**	(1)	100	80.4	81.4	**80.9**	(1)
	PriMeSRL	100	86.8	87.1	87.0(↓0.7)	(3)	100	79.7	80.5	**80.1**(↓0.8)	(1)

Table 7: Comparison of SoTA SRL models with PriMeSRL and CoNLL05 evaluation metrics on CoNLL05 dataset. (r) denotes the ranking of SRL models corresponding to the evaluation metric. **BOLD** shows the best model with CoNLL05 evaluation script and **BOLD** shows the best SRL model with PriMeSRL.

4.3 Evaluation

4.3.1 Predicate sense disambiguation

The PSD column in Table 4 compares the impact of PriMeSRL w.r.t. the existing evaluation script on the EN subset of the CoNLL09 dataset using SoTA SRL models. We observe a consistent quality drop in predicate sense disambiguation (PSD) both for in-domain and out-of-domain (OOD) sets. Surprisingly, we observe a significant quality drop on the OOD set of an average of ∼ 5 F1 points for all the SRL models, which significantly lowers the SoTA performance on the OOD set. This shows that existing SRL models still have a lot of room for improvement.

Continuing the PSD analysis, Table 6 shows example instances from the CoNLL09 dataset that have correct sense numbers (01) but wrong predicate.sense - yet all of which are marked correct by the CoNLL09 evaluation script. For example, the first row shows how the difference between 'stare.01' (*looking intently* [7]) and "look.01" (*causal look* [8]) is ignored. While these two at least share the same underlying meaning (*look*), in row 5 the model's prediction of 'soothe.01' means the opposite of the gold label 'overheat.01' (once again, the existing CoNLL09 evaluation script marks this as correct.) Clearly, predicate sense should be evaluated by including the actual value predicate.sense instead of only relying on the sense number.

4.3.2 Argument head evaluation

Argument classification column in Table 4 compares the impact of PriMeSRL w.r.t the existing evaluation script on the EN subset of the CoNLL09 dataset using SoTA SRL models. We observe a quality drop in the argument classification task both for in-domain and OOD sets, with a significant quality drop of an average of ∼ 8 F1 points on the OOD set. This drop in the argument classification task is expected because part of this error is propagated from the predicate sense disambiguation task which itself is significant. It is interesting to note that, although the major contribution of argument classification drop is due to error propagation from

[7] https://verbs.colorado.edu/propbank/framesets-english-aliases/stare.html

[8] https://verbs.colorado.edu/propbank/framesets-english-aliases/look.html

SRL				Downstream Application		
Input Sentence	SRL model prediction	Existing eval score	PriMeSRL score	Application prompt	Prediction	Expected
[S1] XYZ company bought $2.4 billion in Fannie Mae bonds.	[XYZ company]A0 [bought]sell.01 [$2.4 billion in Fannie Mae bonds]A1	3/3	0/3	QA Who bought Fannie Mae bonds?	None	XYZ company
	[XYZ company]A0 [bought]buy_out.03 [$2.4 billion in Fannie Mae bonds]A1	2/3	0/3	QA Who bought Fannie Mae bonds completely?	XYZ company	None
[S2] XYZ company bought out $2.4 billion in Fannie Mae bonds.	[XYZ company]A0 [bought]buy_out.03 out [$2.4 billion in Fannie Mae bonds]A1	3/3	3/3	NLI Does S2 entails S3?	Yes	No
[S3] XYZ company bought $2.4 billion in Fannie Mae bonds.	[XYZ company]A0 [bought]buy_out.03 [$2.4 billion in Fannie Mae bonds]A1	2/3	0/3			

Table 8: Example illustrations of the how impact of SRL errors on downstream applications is captured by the new evaluation method, where Red color represents the wrong prediction by an SRL model, leading to incorrect predictions by a downstream application (QA: Question Answering; NLI: Natural Language Inference.)

the earlier stage, there is also a consistent drop due to penalizing correct arguments with wrong predicate sense, of $\sim$ 1.5 and $\sim$ 3 F1 points for in-domain and OOD sets, respectively.

Since the performance drop is not uniform, we observe a change in the relative ranking of the SRL models. As an example, the CoNLL09 evaluation script scores the SRL models Blloshmi et al. (2021) and Jindal et al. (2022) similarly (83.0 F1) on OOD set whereas PriMeSRL clearly shows a difference in performance. Further, PriMeSRL makes clear that the quality of existing SRL systems is not as high as previously thought, especially on OOD data.

4.3.3 Argument span evaluation

Similar to argument head evaluation, we compare the impact of PriMeSRL w.r.t to the existing evaluation script on the SRL span dataset (CoNLL05 dataset) using SoTA SRL models in Table 7. Since CoNLL05 does not evaluate predicate sense, we do not observe the impact of incorrect PSD on argument classification. Therefore, the only drop of argument classification is due to incorrect handling of C-X and R-X arguments. Although Table 5 shows that the total number of C-X and R-X in the CoNLL05 dataset is $\sim 5\%$ of the total number of arguments, we only observe a slight drop in quality evaluation ($< 1\%$) with PriMeSRL. This is because on argument span evaluation, PriMeSRL is similar to CoNLL05 (except in a few cases as described in the last row-block of Tables 2 and 3.) As in the comparison with the CoNLL09 dataset, we again observe a change in the relative ranking of the SRL models.

4.4 Discussion

The existing evaluation metrics for SRL are disconnected from the actual practical performance of the SRL models. This makes it difficult to choose the best quality SRL model for the required downstream application. Current evaluation metrics do not pay sufficient attention to the error propagation aspect of the four-staged SRL task; instead, they evaluate the steps independently and linearly combine them to compute the overall SRL system score. However, the analyses in Tables 4 and 7 clearly show that the linear combination of the independent performance of individual steps is not equivalent to the true overall quality.

This does not negate the usefulness of the existing evaluation metrics. Indeed, these metrics provide an evaluation of each individual step, serving as an important guide for improving the quality of individual steps and hence the overall quality of the SRL system. However, whenever a real-world NLP system utilizes an SRL system as one of its components, it is important to understand the quality of semantic roles in relation with, and conditional on their predicate sense disambiguation. Table 8 illustrates the impact of such SRL errors on two downstream applications (question answering and natural language inference). Existing evaluation scripts overlook such SRL errors and treat them as correct, despite the fact that the predicted predicate-argument structure is meaningless and leads to incorrect outputs for the downstream application.

5 Conclusion

In this paper, we highlighted key issues with existing SRL evaluation metrics and showed that the proposed evaluation metric, PriMeSRL, scores SoTA SRL models in a more accurate and intuitive manner. By releasing our evaluation code, we plan to promote these metrics in the community in order to improve the evaluation quality for SRL systems that contribute to downstream applications.

Limitations

We have shown the impact of our proposed new evaluation metrics in the current SoTA SRL models ranking. To further validate the impact of this work, we plan to conduct an in-depth study on how downstream applications' performance relates to the evaluation metrics in future work.

We acknowledge that the problems we have pointed out for previous evaluation metrics are not bugs, but rather design decisions given the timing of the shared tasks and the limitations on datasets and methods. Consider, for instance, that a unified syntactic dependency annotation schema like Universal Dependencies (Nivre et al., 2016) was unavailable before October 2014. Given that, in this paper, we didn't present a deep discussion on the impact of UD compared to previously used syntactic dependencies schemas.

Acknowledgements

We would like to thank our anonymous reviewers for their constructive comments and feedback. Further, the authors acknowledge the IBM Research Cognitive Computing Cluster service for providing resources that have contributed to the research results reported within this paper.

References

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.

Rexhina Blloshmi, Simone Conia, Rocco Tripodi, and Roberto Navigli. 2021. Generating senses and roles: An end-to-end model for dependency-and span-based semantic role labeling. In *IJCAI*, pages 3786–3793.

Blanca Calvo Figueras, Montse Oller, and Rodrigo Agerri. 2022. A semantics-aware approach to automated claim verification. In *Proceedings of the Fifth Fact Extraction and VERification Workshop (FEVER)*, pages 37–48, Dublin, Ireland. Association for Computational Linguistics.

Xavier Carreras and Lluís Màrquez. 2004. Introduction to the CoNLL-2004 shared task: Semantic role labeling. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 89–97, Boston, Massachusetts, USA. Association for Computational Linguistics.

Xavier Carreras and Lluís Màrquez. 2005. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, pages 152–164, Ann Arbor, Michigan. Association for Computational Linguistics.

Simone Conia, Andrea Bacciu, and Roberto Navigli. 2021. Unifying cross-lingual semantic role labeling with heterogeneous linguistic resources. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 338–351, Online. Association for Computational Linguistics.

Simone Conia and Roberto Navigli. 2022. Probing for predicate argument structures in pretrained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4622–4632.

Tomasz Dryjański, Monika Zaleska, Bartek Kuźma, Artur Błażejewski, Zuzanna Bordzicka, Paweł Bujnowski, Klaudia Firlag, Christian Goltz, Maciej Grabowski, Jakub Jończyk, Grzegorz Kłosiński, Bartłomiej Paziewski, Natalia Paszkiewicz, Jarosław Piersa, and Piotr Andruszkiewicz. 2022. Samsung research Poland (SRPOL) at SemEval-2022 task 9: Hybrid question answering using semantic roles. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1263–1273, Seattle, United States. Association for Computational Linguistics.

Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.

Shaik Fharook, Syed Sufyan Ahmed, Gurram Rithika, Sumith Sai Budde, Sunil Saumya, and Shankar Biradar. 2022. Are you a hero or a villain? a semantic role labelling approach for detecting harmful memes. In *Proceedings of the Workshop on Combating Online Hostile Posts in Regional Languages during Emergency Situations*, pages 19–23, Dublin, Ireland. Association for Computational Linguistics.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, pages 1–18, Boulder, Colorado. Association for Computational Linguistics.

Ishan Jindal, Ranit Aharonov, Siddhartha Brahma, Huaiyu Zhu, and Yunyao Li. 2020. Improved semantic role labeling using parameterized neighborhood memory adaptation. *arXiv preprint arXiv:2011.14459*.

Ishan Jindal, Alexandre Rademaker, Michał Ulewicz, Ha Linh, Huyen Nguyen, Khoi-Nguyen Tran, Huaiyu Zhu, and Yunyao Li. 2022. Universal proposition bank 2.0. In *Proceedings of the Language Resources and Evaluation Conference*, pages 1700–1711, Marseille, France. European Language Resources Association.

Daniel Jurafsky and James H. Martin. 2021. Speech and language processing. `https://web.stanford.edu/~jurafsky/slp3/`.

Paul Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA).

Oleksandr Kolomiyets, Parisa Kordjamshidi, Marie-Francine Moens, and Steven Bethard. 2013. SemEval-2013 task 3: Spatial role labeling. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 255–262, Atlanta, Georgia, USA. Association for Computational Linguistics.

Parisa Kordjamshidi, Steven Bethard, and Marie-Francine Moens. 2012. SemEval-2012 task 3: Spatial role labeling. 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)*, pages 365–373, Montréal, Canada. Association for Computational Linguistics.

Zuchao Li, Shexia He, Jiaxun Cai, Zhuosheng Zhang, Hai Zhao, Gongshen Liu, Linlin Li, and Luo Si. 2018. A unified syntax-aware framework for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2401–2411, Brussels, Belgium. Association for Computational Linguistics.

Ken Litkowski. 2004. Senseval-3 task: Automatic labeling of semantic roles. In *Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 9–12, Barcelona, Spain. Association for Computational Linguistics.

Ling Liu, Ishan Jindal, and Yunyao Li. 2022. Is semantic-aware bert more linguistically aware? a case study on natural language inference. In *Proceedings of the Workshop on Structured and Unstructured Knowledge Integration (SUKI)*, Seattle, USA. Association for Computational Linguistics.

Edward Loper, Szu-Ting Yi, and Martha Palmer. 2007. Combining lexical resources: mapping between propbank and verbnet. In *Proceedings of the 7th International Workshop on Computational Linguistics, Tilburg, the Netherlands*.

Umar Maqsud, Sebastian Arnold, Michael Hülfenhaus, and Alan Akbik. 2014. Nerdle: Topic-specific question answering using wikia seeds. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*, pages 81–85.

Marco Marelli, Luisa Bentivogli, Marco Baroni, Raffaella Bernardi, Stefano Menini, and Roberto Zamparelli. 2014. SemEval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 1–8, Dublin, Ireland. Association for Computational Linguistics.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2018. A survey on open information extraction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3866–3878, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association (ELRA).

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Sameer Pradhan, Julia Bonn, Skatje Myers, Kathryn Conger, Tim O'Gorman, James Gung, Kristin Wright-Bettner, and Martha Palmer. 2022. Propbank comes of age—larger, smarter, and more diverse. In *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, pages 278–288.

Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 2007. SemEval-2007 task-17: English lexical sample, SRL and all words. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 87–92, Prague, Czech Republic. Association for Computational Linguistics.

Reinhard Rapp. 2022. Using semantic role labeling to improve neural machine translation. In *Proceedings of the Language Resources and Evaluation Conference*, pages 3079–3083, Marseille, France. European Language Resources Association.

Michael Roth and Mirella Lapata. 2016. Neural semantic role labeling with dependency path embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1192–1202, Berlin, Germany. Association for Computational Linguistics.

Karin Kipper Schuler. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. University of Pennsylvania.

Chen Shi, Shujie Liu, Shuo Ren, Shi Feng, Mu Li, Ming Zhou, Xu Sun, and Houfeng Wang. 2016. Knowledge-based semantic embedding for machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2245–2254.

Peng Shi and Jimmy Lin. 2019. Simple bert models for relation extraction and semantic role labeling. *arXiv preprint arXiv:1904.05255*.

Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The CoNLL 2008 shared task on joint parsing of syntactic and semantic dependencies. In *CoNLL 2008: Proceedings of the Twelfth Conference on Computational Natural Language Learning*, pages 159–177, Manchester, England. Coling 2008 Organizing Committee.

Wen-tau Yih, Matthew Richardson, Chris Meek, Ming-Wei Chang, and Jina Suh. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–206.

Hongming Zhang, Haoyu Wang, and Dan Roth. 2020a. Unsupervised label-aware event trigger and argument classification. *CoRR*, abs/2012.15243.

Yu Zhang, Qingrong Xia, Shilin Zhou, Yong Jiang, Zhenghua Li, Guohong Fu, and Min Zhang. 2021. Semantic role labeling as dependency parsing: Exploring latent tree structures inside arguments. *arXiv preprint arXiv:2110.06865*.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020b. Semantics-aware bert for language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9628–9635.

A Some historical background to existing evaluation metrics for SRL

Shared tasks have boosted the development of systems capable of extracting predicates and arguments from natural language sentences. Two regular academic events have promoted SRL shared tasks: SemEval and CoNLL. In this section, we summarize the approaches to SRL evaluation in the shared tasks and categorize their shortcomings.

A.1 Senseval and SemEval

SemEval (Semantic Evaluation) is a series of evaluations of computational semantic analysis systems that evolved from the Senseval (word sense evaluation) series. The SENSEVAL-3 (Litkowski, 2004) was about the automatic labeling of semantic roles and was designed to encourage research into and use of the FrameNet dataset. The systems receive as input unsegmented sentences (the constituents are not identified) a target word and its frame. They have to identify the frame elements within that sentence and tag them with the appropriate frame element name. In general, FrameNet frames contain many frame elements (an average of 10), most of which are not instantiated in a given sentence. Systems were not penalized if they returned more frame elements than those identified in the gold data. In scoring, each frame element returned by a system was counted as an item attempted. If the frame element had been identified in the gold data, the answer was scored as correct. In addition, the scoring program required that the frame boundaries identified by the system's answer overlap with the gold annotation. An additional measure of system performance was the degree of overlap. If a system's answer coincided precisely with the start and end position in the gold data, the system received an overlap score of 1.0. If not, the overlap score was the number of characters overlapping divided by the length of the gold annotation. The number attempted was the number of non-null frame elements generated by a system. Precision was computed as the number of correct answers divided by the number attempted. The recall was computed as the number of correct answers divided by the number of frame elements in the test set. Overlap was the average overlap of all correct answers. The percent Attempted was the number of frame elements generated divided by the number of frame elements in the test set, multiplied by 100.

At SemEval-2007, three tasks evaluate SRL. In task 17 (subtask 2), the goal of the systems was to locate the constituents, which are the arguments of a given verb, and assign them appropriate semantic roles. Systems have to annotate the corpus using two different role label sets: the PropBank and the VerbNet. SemLink mapping (Loper et al., 2007) was used to generate the VerbNet roles. The precision, recall, and F-measure for both role label sets were calculated for each system output using the `srl-eval.pl` script from the CoNLL-2005 scoring package (Carreras and Màrquez, 2005) (see below). Task 18 focused on Arabic and also used the same CoNLL-2005 scoring package. Task 19 consists of recognizing words and phrases that evoke semantic frames from FrameNet and their semantic dependents, which are usually, but not always, their syntactic dependents. The evaluation measured precision and recall for frames and frame elements, with partial credit for incorrect but closely related frames. Two types of evaluation were carried out. The first is the label matching evaluation. The participant's labeled data were compared directly with the gold standard labeled using the same evaluation procedure used in the previous SRL tasks at SemEval. The second is the semantic dependency evaluation, in which both the gold standard and the submitted data were first converted to semantic dependency graphs and compared.

SemEval-2012 and SemEval-2013 introduced the 'Spatial Role Labeling' task. It concerns the identification of trajectors, landmarks, spatial indicators, the links between them, and the type of spatial relationships, including region, direction, and distance. Although similar to the standard SRL task, we will not discuss Spatial Role Labeling and its evaluation in this paper. Starting from SemEval-2014, a deeper semantic representation of sentences in a single graph-based structure via semantic parsing substituted the 'shallow' SRL tasks.

A.2 CoNLL

The Conference on Computational Natural Language Learning (CoNLL) is a yearly conference organized by the ACL's Special Interest Group on Natural Language Learning (SIGNLL), focusing on theoretically, cognitively and scientifically motivated approaches to computational linguistics since 1999. The 2004 and 2005 shared tasks of the CoNLL were dedicated to SRL monolingual setting (English). The CoNLL-2008 shared task proposes a unified dependency-based formalism, which models both syntactic dependencies and semantic roles. The CoNLL-2009 builds on the CoNLL-2008 task and extends it to multiple languages.

The CoNLL-2004 shared task (Carreras and Màrquez, 2004) was based on the PropBank corpus, six sections of the Wall Street Journal part of the Penn Treebank (Kingsbury and Palmer, 2002) enriched with predicate–argument structures. The participants need to come up with machine learning strategies to SRL on the basis of only partial syntactic information, avoiding the use of full parsers and external lexico-semantic knowledge bases. The annotations provided for the development of systems include, apart from the argument boundaries and role labels, the levels of processing treated in the previous editions of the CoNLL shared task, i.e., words, PoS tags, base chunks, clauses, and named entities. In practice, number of target verbs are marked in a sentence, each governing one proposition. A system has to recognize and label the arguments of each target verb. The systems were evaluated with respect to precision, recall and the F1 measure using the `srl-eval.pl` program. For an argument to be correctly recognized, the words spanning the argument as well as its semantic role have to be correct. The verb argument is the lexicalization of the predicate of the proposition. Most of the time, the verb corresponds to the target verb of the proposition, which is provided as input, and only in few cases the verb participant spans more words than the target verb. This situation makes the verb easy to identify and, since there is one verb with each proposition, evaluating its recognition overestimates the overall performance of a system. For this reason, the verb argument is excluded from evaluation. The shared task proceedings does not details how non-continuous arguments are evaluated.

Compared to the shared task of CoNLL-2004, three novelties were introduced in the 2005 edition

(Carreras and Màrquez, 2005): 1) the complete syntactic trees, with information of the lexical head for each syntactic constituent, given by two alternative parsers have been provided as input; 2) the training corpus has been substantially enlarged; 3) a cross-corpora evaluation is performed using a fresh test set from the Brown corpus. Evaluation didn't changed compared to CoNLL-2004 and it was reported to use the same evaluation code, a system has to recognize and label the arguments of each target verb. To support the role labeling task, sentences contain input annotations, that consist of syntactic information and named entities. Evaluation is performed on a collection of unseen test sentences, that are marked with target verbs and contain only predicted input annotations.

The CoNLL 2008 shared task (Surdeanu et al., 2008) was dedicated to the joint parsing of syntactic and semantic dependencies. The shared task was divided into three subtasks: (i) parsing of syntactic dependencies, (ii) identification and disambiguation of semantic predicates, and (iii) identification of arguments and assignment of semantic roles for each predicate. SRL was performed and evaluated using a dependency-based representation for both syntactic and semantic dependencies.

The task addressed propositions centered around both verbal and nominal predicates. The data was composed by the Penn Treebank, BBN's named entity corpus, PropBank and NomBank. The dependency-annotated data was obtain from a conversion algorithm from the constituent analyses. convert the underlying constituent analysis of PropBank and NomBank into a dependency analysis, the head of a semantic argument was identified with a straightforward heuristic. But there are cases that require special treatment, some arguments ended up with several syntactic heads and some arguments that were initially discontinuous in PropBank or NomBank where merged.

The official evaluation measures consist of three different scores: (i) syntactic dependencies are scored using the labeled attachment score (LAS), (ii) semantic dependencies are evaluated using a labeled F1 score, and (iii) the overall task is scored with a macro average of the two previous scores. The semantic propositions are evaluated by converting them to semantic dependencies, i.e., a semantic dependency from every predicate to all its individual arguments were created. These dependencies are labeled with the labels of the corresponding arguments. Additionally, a semantic dependency from each predicate to a virtual ROOT node was created. The latter dependencies are labeled with the predicate senses. This approach guarantees that the semantic dependency structure conceptually forms a single-rooted, connected (not necessarily acyclic) graph. More importantly, this scoring strategy implies that if a system assigns the incorrect predicate sense, it still receives some points for the arguments correctly assigned. Several additional evaluation measures were applied to further analyze the performance of the participating systems. The *Exact Match* reports the percentage of sentences that are completely correct, i.e., all the generated syntactic dependencies are correct and all the semantic propositions are present and correct. The *Perfect Proposition F1* score entire semantic frames or propositions. The ratio between labeled F1 score for semantic dependencies and the LAS for syntactic dependencies.

As in CoNLL-2008, the CoNLL-2009 shared task (Hajič et al., 2009) combined syntactic dependency parsing and the task of identifying and labeling semantic arguments of verbs or nouns for six more languages (Catalan, Chinese, Czech, German, Japanese and Spanish) in addition to the original English from CoNLL-2008. Participants can choose the joint task (syntactic dependency parsing and SRL), or SRL-only (syntactic dependency provided). The novelty is that the evaluation data indicated which words were to be dealt with (for the SRL task). Predicate disambiguation was still part of the task, whereas the identification of argument-bearing words was not. This decision was made to compensate for the significant differences between languages and between the annotation schemes used. The evaluation of SRL was done similar to CoNLL-2008.