Generating Senses and RoLes: An End-to-End Model for Dependency- and Span-based Semantic Role Labeling

Rexhina Blloshmi , **Simone Conia** , **Rocco Tripodi** and **Roberto Navigli**
Sapienza NLP Group, Sapienza University of Rome, Italy
{rexhina.blloshmi, simone.conia, rocco.tripodi, roberto.navigli}@uniroma1.it

Abstract

Despite the recent great success of the sequence-to-sequence paradigm in Natural Language Processing, the majority of current studies in Semantic Role Labeling (SRL) still frame the problem as a sequence labeling task. In this paper we go against the flow and propose GSRL (*Generating Senses and RoLes*), the first sequence-to-sequence model for end-to-end SRL. Our approach benefits from recently-proposed decoder-side pretraining techniques to generate both sense and role labels for all the predicates in an input sentence at once, in an end-to-end fashion. Evaluated on standard gold benchmarks, GSRL achieves state-of-the-art results in both dependency- and span-based English SRL, proving empirically that our simple generation-based model can learn to produce complex predicate-argument structures. Finally, we propose a framework for evaluating the robustness of an SRL model in a variety of synthetic low-resource scenarios which can aid human annotators in the creation of better, more diverse, and more challenging gold datasets. We release GSRL at `github.com/SapienzaNLP/gsrl`.

1 Introduction

Semantic Role Labeling (SRL) is commonly referred to as the task of automatically addressing the question "Who did What, to Whom, Where, When, and How?" [Gildea and Jurafsky, 2002; Màrquez *et al.*, 2008]. More specifically, the task consists in: i) detecting the utterances, called predicates, that express an event or convey an action; ii) identifying the sentential constituents, called arguments, that participate in the event or action outlined by each predicate; and finally, iii) choosing the most appropriate relation, called semantic role, that governs each predicate-argument pair. The resulting structured output can be seen as a form of shallow semantic parsing and, therefore, achieving human parity in SRL is often regarded as a fundamental step towards Natural Language Understanding [Navigli, 2018]. Unsurprisingly, SRL has garnered increasing attention through time, and numerous studies have found it to be beneficial in a wide range of downstream applications, not only in Natural Language Processing but also in Computer Vision, including: Question Answering [Shen and Lapata, 2007], Machine Translation [Marcheggiani *et al.*, 2018], Visual Semantic Role Labeling [Gupta and Malik, 2015] and Situation Recognition [Yatskar *et al.*, 2016].

Over the years, researchers made a great many steps forward in the design of better SRL models, moving from manually-engineered feature templates to multilayered neural networks [Cai *et al.*, 2018; Marcheggiani and Titov, 2020], and from static to dynamically-contextualized word representations [He *et al.*, 2019; Conia and Navigli, 2020]. While past and present studies have accomplished impressive results, the vast majority of the state-of-the-art models proposed year after year have framed SRL as a sequence labeling task [Cai *et al.*, 2018; Li *et al.*, 2019], and only a small handful of studies have put forward SRL systems based on sequence-to-sequence learning [Sutskever *et al.*, 2014], despite the growing success of this paradigm in other areas of Natural Language Understanding [Yin *et al.*, 2016; Lewis *et al.*, 2020; Raffel *et al.*, 2020]. Currently, SRL sequence-to-sequence models fall behind traditional sequence labeling approaches in terms of performance [Daza and Frank, 2018] and can address only a portion of the SRL pipeline [Daza and Frank, 2019], making them an unappealing option for downstream applications.

In this paper, we aim at addressing these issues and propose GSRL (*Generating Senses and RoLes*), a novel approach to generating both predicate senses and semantic roles. The contributions of our work are manifold:

- We introduce the first sequence-to-sequence model for end-to-end SRL, tackling predicate sense disambiguation, argument identification and argument classification as a single generation task;
- We demonstrate that sequence-to-sequence learning can achieve state-of-the-art results, previously attained only by sequence labeling approaches, in multiple gold benchmarks for both dependency- and span-based English SRL;
- We compare different strategies to represent predicate-argument relations and generate structured, graph-like sense and role annotations, analyzing their positives and negatives;
- Motivated by the convergence in the performance of recent SRL systems, we propose a framework to i) evalu-

ate future innovations in more challenging settings and ii) aid the creation of new SRL datasets.

2 Related Work

Dependency and Span-based SRL. SRL is traditionally framed as either a dependency-based [Surdeanu *et al.*, 2008; Hajic *et al.*, 2009] or a span-based [Carreras and Màrquez, 2005; Pradhan *et al.*, 2012] labeling task. Given a predicate in a sentence, the difference between the two settings is in the formalism used to represent its arguments. As shown in Fig. 1, span-based SRL requires the identification and classification of the entire textual span of an argument, whereas dependency-based SRL is concerned about labeling only the head of the argument. Even if, to date, it is not clear whether one is better than the other [Li *et al.*, 2019], researchers tend to agree that the two formalisms pose different challenges and capture complementary aspects of the overall task [Zhou *et al.*, 2020]. Our work encompasses both span- and dependency-based SRL, demonstrating that a generation-based approach is able to achieve state-of-the-art results in both.

Sequence-to-Sequence SRL. Sequence-to-sequence learning was introduced as a general approach to sequence learning that makes minimal assumptions on the sequence structure [Sutskever *et al.*, 2014]. While it was initially conceived for Machine Translation [Bahdanau *et al.*, 2015], sequence-to-sequence learning rapidly found success in a variety of Natural Language Processing tasks from Question Answering [Yin *et al.*, 2016] to Dialogue [Song *et al.*, 2019], Text Generation [Lewis *et al.*, 2020; Raffel *et al.*, 2020] and more recently Semantic Parsing [Blloshmi *et al.*, 2020; Bevilacqua *et al.*, 2021; Procopio *et al.*, 2021], *inter alia*. Recent work in SRL, however, still revolves predominantly around sequence labeling approaches [Cai and Lapata, 2019b; Xia *et al.*, 2019; Conia and Navigli, 2020; Marcheggiani and Titov, 2020; Conia *et al.*, 2021], with only a small handful of attempts at tackling the task in a sequence-to-sequence fashion. Daza and Frank [2018] and Daza and Frank [2019] are, to the best of our knowledge, the most notable studies on generation-based models for SRL, but their performance on standard benchmarks lags behind state-of-the-art sequence labeling techniques. Nevertheless, inspired by recent advances in sequence-to-sequence paradigm and innovative decoder-side pretraining [Lewis *et al.*, 2020], we show that our *sequence-to-sequence* model is able to challenge *sequence labeling* systems across multiple gold benchmarks.

End-to-End SRL. Due to its complexity, SRL is often divided into a pipeline of four stages or subtasks: predicate detection, predicate disambiguation, argument identification and argument classification. While early work tried to develop distinct systems for each subtask, later studies successfully demonstrated that sequence labeling models [Cai *et al.*, 2018; Li *et al.*, 2019] can benefit from tackling some of these tasks jointly with multitask learning [Caruana, 1997]. However, sequence-to-sequence models proposed over the last few years can only solve the later stages of the SRL pipeline – namely, argument identification and argument classification – and, therefore, they still require an underlying system to perform at least predicate sense disambiguation [Daza and Frank, 2018; Daza and Frank, 2019]. Indeed, the function of a semantic role is often well-defined only with respect to a given predicate sense, especially when dealing with PropBank-like predicate-argument structure inventories [Palmer *et al.*, 2005]. For example, even though there are two `ARG1` role labels in Fig. 1, they actually encode different relations: when `ARG1` is associated with the predicate sense `tell.01`, it refers to the *utterance* or *topic* of the action, whereas, when it is an argument for the predicate sense `take.01`, it refers to the *thing taken* or *theme* of the action. While predicate sense disambiguation is essential to SRL, introducing structured predicate-argument relations in a sequence-to-sequence model is not trivial. In our work, we explore different predicate-argument linearization schemes and introduce, to the best of our knowledge, the first end-to-end sequence-to-sequence model to successfully generate both sense and role labels.

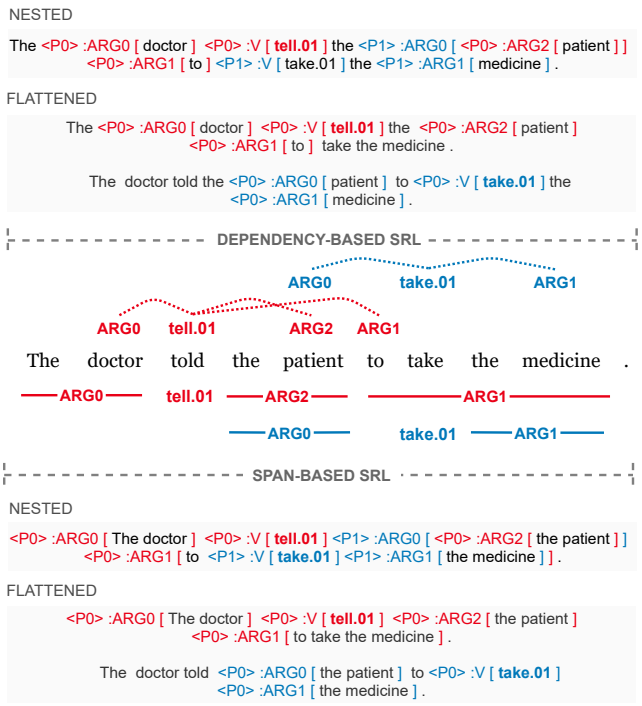


Figure 1: Example of a sentence with two predicates: dependency-based SRL (upper part) and span-based SRL (lower part) and their corresponding *nested* and *flattened* linearizations.

3 Methodology

3.1 SRL as a Sequence-to-Sequence Task

We revisit the *sequence-to-sequence* formulation by Daza and Frank [2018] for PropBank-based SRL and put forward a generalized formulation that is able to handle not only semantic role labels but also predicate sense labels. Formally, given a sentence $\mathbf{s} = \langle w_1, w_2, \ldots, w_{|\mathbf{s}|} \rangle$ where each word w_i belongs to either the vocabulary of words V^{W} or a vocabulary of special tokens V^{ST}, the model is required to generate a se-

quence $\mathbf{o} = \langle o_1, o_2, \ldots, o_{|\mathbf{o}|} \rangle$ where each token o_i belongs to either the input sentence $\mathbf{s}$, the vocabulary of special tokens V^{ST}, the semantic role vocabulary V^{SR}, or the predicate sense vocabulary V^{PS}.

As shown in Fig. 1, we propose two strategies for generating the predicate-argument relations:

- *Flattened* linearization in which the model is required to generate a separate sequence $\mathbf{o}_p$ for each predicate p in $\mathbf{s}$, where $\mathbf{o}_p$ contains the sense and role labels only for p;
- *Nested* linearization in which the model is required to generate a single sequence $\mathbf{o}$ containing the sense and role labels for all the predicates in $\mathbf{s}$.

If we exclude predicate sense labels from the generated sequence $\mathbf{o}$, our *flattened* linearization strategy is similar to that of Daza and Frank [2018] and can be considered as a simplified or "unrolled" semantic structure of our *nested* linearization. We argue that the semantics of the *nested* linearization, while being more complex to learn, comes with the advantage of providing the entire predicate-argument structure of the input sentence $\mathbf{s}$ at once, reducing the overhead of generating a number of output sequences equal to the number of predicates in $\mathbf{s}$, and thus being more practical for an end system.

3.2 The GSRL Model

Given the above definition of sequence-to-sequence SRL, we formally frame the task as a conditional generation problem in which we want to maximize the probability $P(\mathbf{g}|\mathbf{t})$ of generating the tokenization $\mathbf{g} = \langle g_1, g_2, \ldots, g_i, \ldots, g_{|\mathbf{g}|} \rangle$ of the output linearization $\mathbf{o}$ conditioned on the tokenization $\mathbf{t} = \langle t_1, t_2, \ldots, t_{|\mathbf{t}|} \rangle$ of the input sentence $\mathbf{s}$:

$$P(\mathbf{g}|\mathbf{t}) = \prod_{i=2}^{|\mathbf{g}|} P(g_i \mid \mathbf{g}_{1:i-1}, \mathbf{t}) \quad (1)$$

where g_1 is the artificially added start token <s>, g_i is the i-th element (token, special token, sense, or role) of the generated output sequence $\mathbf{g}$ and $\mathbf{g}_{1:i-1} = \langle g_1, g_2, \ldots, g_{i-1} \rangle$. Therefore, the probability $P(\mathbf{g}|\mathbf{t})$ of the linearized predicate-argument structure $\mathbf{g}$ for the given sentence $\mathbf{t}$ is computed as the product of the probability of generating each token g_i of $\mathbf{g}$ in an autoregressive fashion.

The GSRL model architecture builds on top of BART [Lewis *et al.*, 2020], a recently proposed denoising autoencoder for sequence-to-sequence learning. BART can be seen as a generalization of several modern language models from BERT (due to the bidirectional encoder) to GPT (with the left-to-right decoder), and it was found to be particularly effective in a wide range of Natural Language Understanding tasks, including tasks that involve complex structured outputs such as semantic parsing [Bevilacqua *et al.*, 2021]. Following BART, our model architecture is based on a Transformer-based neural machine translation architecture [Vaswani *et al.*, 2017], with 12 stacked Transformer layers for both the encoder and the decoder. However, rather than training GSRL to learn to maximize the conditional probability shown in Equation 1 from scratch, we warm-start the model with the weights of BART, which brings two significant advantages. First, GSRL inherits the capability of BART to denoise artificially-corrupted sentences and generate an output sequence that, while (partially) overlapping with the input sequence, can have a different length. This is beneficial to our setting, since the input sequence fed into the model can be seen as a corrupted sentence where the sense and role annotations have been removed. Second, GSRL can take advantage of the world of knowledge coming from the massive amounts of text BART has been pretrained on. Indeed, the original training corpus for BART is composed of five English-language corpora of varying sizes and domains, containing books, stories, news, web content and Wikipedia articles, and thus providing a wealth of information that could otherwise be missing from standard SRL datasets, given their relatively small size.

Vocabulary. We start from the vocabulary of BART which, thanks to its BPE tokenization, includes V^{W}, and extend it by adding i) the set V^{PS} of PropBank predicate sense labels, e.g., `tell.01` and `take.01`, ii) the set V^{SR} of PropBank semantic role labels, e.g., `:ARG0` and `:ARGM-NEG`, and iii) the set V^{ST} of special tokens to distinguish between verbal and nominal predicates, i.e., `:V` and `:N` respectively, and to identify the predicates in the sentence, i.e., <P*i*>, where i is the order of the predicate in the input sentence from left to right. At input level, we make sure that the BPE tokenizer does not split the additional tokens. Therefore, adding these task-specific atomic tokens to the vocabulary allows for a more compact linearized SRL structure. Finally, we randomly initialize the embeddings of the additional tokens and update their values during training.

Preprocessing. The input sentence is preprocessed differently depending on the linearization strategy – *flattened* or *nested* – chosen to train the GSRL model. Before feeding an input sentence into the model, we indicate each predicate with a special token <P*i*> which guides the model towards learning to distinguish between different predicates and to specifically generate the argument roles for each of them, where $i = 0$ in the *flattened* linearization and $0 \leq i < n_p$ in the *nested* linearization, with n_p being the total number of predicates in the sentence. In the flattened linearization setting, the input sentence is repeated n_p times, e.g., for the example in Fig. 1 the input sentence would be preprocessed twice: i) "The doctor <P0> `:V` [told] the patient to take the medicine", and ii) "The doctor told the patient to <P0> `:V` [take] the medicine". When the GSRL model is trained to generate *nested* linearizations, the input sentence is preprocessed to indicate all the predicates at once, e.g., "The doctor <P0> `:V` [told] the patient to <P1> `:V` [take] the medicine".

Postprocessing. As opposed to sequence labeling approaches, our sequence-to-sequence model is not only trained to produce sense and role labels, but also to autoregressively regenerate the words of the input sentence. Therefore, an output sequence is valid only if the following two conditions are met: i) its words can be aligned to the words of the input sequence, and ii) all the predicate-argument structures follow the PropBank annotation guidelines. Indeed, in order to enforce valid predicate-argument structures during the annotation process, PropBank-based SRL requires human annotators to follow a set of guidelines which state that core roles

(ARG0, ARG1, etc.) must appear at most once for each predicate, two arguments of the same predicate must not overlap, reference roles (R-ARG0, R-ARGM-TMP, etc.) can only appear if they refer to an existing core role in the sentence, and continuation roles (C-ARG0, C-ARGM-TMP, etc.) can only appear after the core role they refer to, *inter alia*. For the sake of simplicity, our model is not explicitly constrained to generate a valid predicate-argument structure, and we only adopt the following simple heuristics to postprocess an output sequence:

- In span-based SRL, we close at most one unenclosed argument span, positioning the closing bracket so that there are no two overlapping arguments for the same predicate;
- In span-based SRL, if more than one span is unenclosed, we discard all the spans;
- In both dependency- and span-based SRL, if two arguments of the same predicate overlap, we discard all the arguments for the sentence.

Previous studies have shown that explicitly enforcing PropBank constraints leads to more accurate predictions [Li *et al.*, 2019], but in this work we focus on unconstrained generation and leave constrained generation for future work.

4 Experiments

4.1 Evaluation Benchmarks

We train and evaluate GSRL on the standard splits of the English datasets provided as part of the CoNLL-2009 [Hajic *et al.*, 2009] and CoNLL-2012 [Pradhan *et al.*, 2012] shared tasks, which rapidly became two standard benchmarks for dependency- and span-based SRL, respectively. While CoNLL-2009 is mainly composed of finance-related documents coming from the Wall Street Journal, CoNLL-2012 is a varied collection of news, conversations and magazine articles. Additionally, CoNLL-2009 includes an out-of-domain test set containing excerpts from the Brown Corpus.

Data Statistics. We define the semantic complexity of a dataset as the number of predicate-argument relations that appear in each sentence on average. In CoNLL-2012, we observe that around 70% of the sentences are annotated with at most 5 role labels and 3 predicates, with an average of 2.8 predicates per sentence. However, this is not the case in CoNLL-2009 where only 20% of the sentences contain at most 5 role labels, and only 40% feature at most 3 predicates. In fact, CoNLL-2009 has an average of 4.7 predicates per sentence, almost twice the number compared to CoNLL-2012. These statistics suggest that the semantic complexity of CoNLL-2009 is higher than that of CoNLL-2012, and thus it is to be expected that the predicate-argument structures in CoNLL-2009 should be more complex, making the *nested* linearizations deeper and more difficult to learn.

Evaluation Metrics. In the following Sections, we report the scores of the official scorers provided as part of the CoNLL shared tasks to measure the performance of a participating system. More specifically, the standard evaluation script for span-based PropBank-style SRL is the CoNLL-2005 scorer[1] which computes precision, recall and F1 score of the semantic roles. For dependency-based PropBank-style SRL we use the CoNLL-2009 scorer[2] which takes into account both sense and role labels to compute what is referred to as "semantic" precision and recall:

$$P_{\text{SEM}} = (\text{TP}^{\text{pred}} + \text{TP}^{\text{role}})/(\text{N}^{\text{pred}} + \text{TP}^{\text{role}} + \text{FP}^{\text{role}})$$
$$R_{\text{SEM}} = (\text{TP}^{\text{pred}} + \text{TP}^{\text{role}})/(\text{N}^{\text{pred}} + \text{TP}^{\text{role}} + \text{FN}^{\text{role}})$$

where TP, FP and FN are the true positives, false positives and false negatives, respectively, while N^{pred} is the total number of predicates.

PARAMETER	PICK	SEARCH SPACE
LR	$5 * 10^{-5}$	1/5/10/50 $*10^{-5}$
Betas	0.9, 0.999	-
Epochs	20	[10, 20]
Dropout	0.25	0.1 to 0.25, (0.05)
W. Decay	0.004	0.001 to 0.01, (+0.001)
LR sched.	constant	-
Grad. accum.	10	[1, 5, 10, 15, 20]

Table 1: GSRL hyperparamter values and search space.

4.2 Training and Tuning

We train different model configurations using the *flattened* and *nested* linearizations, $\text{GSRL}_{\textit{flattened}}$ and $\text{GSRL}_{\textit{nested}}$ hereafter. For both variants, their weights are warm-started using $\text{BART}_{\text{large}}$ (406M parameters) from the Transformers library.[3] Differently from vanilla BART, we increase the dropout rate between the Transformer layers from 0.1 to 0.25 and we do not penalize the model for the generation of repeated *ngrams*, e.g., multiple closing brackets. In Table 1 we report the hyperparameters space of GSRL. We pick the parameters using random search with 5 trials in the search space indicated in the third column. Finally, we select the best model based on its F1 score on the development dataset. At prediction time we perform only greedy decoding, since beam searching did not show improvements in our preliminary experiments. Each GSRL model is trained for 20 epochs with a batch size of 800 tokens, using the RAdam [Liu *et al.*, 2020] optimizer with a fixed learning rate of 1×10^{-5} and gradient accumulation every 10 batches. The training process is carried out on a single GPU (Nvidia GeForce GTX 1080Ti): $\text{GSRL}_{\textit{flattened}}$ requires 30 and 40 hours of training time on CoNLL-2009 and CoNLL-2012, respectively, while $\text{GSRL}_{\textit{nested}}$ requires 11 and 20 hours on CoNLL-2009 and CoNLL-2012, respectively.

4.3 Comparison Systems

The vast majority of the recent advances in SRL come from sequence labeling approaches, which currently represent the state of the art in both span- and dependency-based SRL. Therefore, we mainly compare our sequence-to-sequence

[1] cs.upc.edu/∼srlconll/soft.html

[2] ufal.mff.cuni.cz/conll2009-st/scorer.html

[3] huggingface.co/transformers/model_doc/bart.html

CoNLL-2009 – In domain	P	R	F_1
Sequence labeling models			
Cai and Lapata [2019b]	90.9	89.1	90.0
Lyu *et al.* [2019]	–	–	90.1
Kasai *et al.* [2019]	90.3	90.0	90.2
Li *et al.* [2019]	89.6	91.2	90.4
He *et al.* [2019]	90.4	91.3	90.9
Chen *et al.* [2019]	90.7	91.4	91.1
Cai and Lapata [2019a]	91.7	90.8	91.2
Shi and Lin [2019]	92.4	92.3	92.4
Conia and Navigli [2020] $_{\text{XLM-R}}$	92.2	92.6	92.4
Conia and Navigli [2020] $_{\text{BERT}}$	92.5	92.7	92.6
Sequence-to-sequence models			
Daza and Frank [2019]	–	–	90.8
GSRL$_{\textit{nested}}$	91.8	86.5	89.0
GSRL$_{\textit{flattened}}$	92.9	92.0	92.4

Table 2: Results on the English in-domain test set of the CoNLL-2009 task for dependency-based SRL. P: precision. R: recall.

CoNLL-2009 – Out of domain	P	R	F_1
Sequence labeling models			
Li *et al.* [2019]	–	–	81.5
Lyu *et al.* [2019]	–	–	82.2
Chen *et al.* [2019]	–	–	82.7
Conia and Navigli [2020] $_{\text{XLM-R}}$	–	–	85.2
Conia and Navigli [2020] $_{\text{BERT}}$	–	–	85.9
Sequence-to-sequence models			
Daza and Frank [2019]	–	–	84.1
GSRL$_{\textit{nested}}$	85.0	80.1	82.5
GSRL$_{\textit{flattened}}$	85.8	84.5	85.2

Table 3: Results on the English out-of-domain test of the CoNLL-2009 task for dependency-based SRL. P: precision. R: recall.

model against the recent innovations proposed by such sequence labeling models, namely, jointly learning SRL and syntax [Cai and Lapata, 2019b], iteratively refining the output SRL labels [Lyu *et al.*, 2019], devising a set of syntactic "supertags" [Kasai *et al.*, 2019], integrating syntactic rules into the learning process [He *et al.*, 2019], learning predicate-argument interactions through capsule networks [Chen *et al.*, 2019], better exploiting the knowledge of language models [Shi and Lin, 2019; Conia and Navigli, 2020], and modeling syntactic dependencies with graph convolutions [Marcheggiani and Titov, 2020]. Also, we compare with Daza and Frank [2018; 2019], who proposed, to the best of our knowledge, the currently best-performing sequence-to-sequence models for SRL, with the important difference that their architectures i) are not able to handle multiple predicates at once, and ii) do not address predicate sense disambiguation, i.e., they are not end-to-end (see §2, End-to-End SRL).

CoNLL-2012	P	R	F_1
Sequence labeling models			
Ouchi *et al.* [2018]	87.1	85.3	86.2
Li *et al.* [2019]	85.7	86.3	86.0
Shi and Lin [2019]	85.9	87.0	86.5
Marcheggiani and Titov [2020]	86.5	87.1	86.8
Conia and Navigli [2020]	86.9	87.7	87.3
Sequence-to-sequence models			
Daza and Frank [2018]	–	–	75.4
GSRL$_{\textit{nested}}$	87.1	86.6	86.8
GSRL$_{\textit{flattened}}$	87.8	86.8	87.3

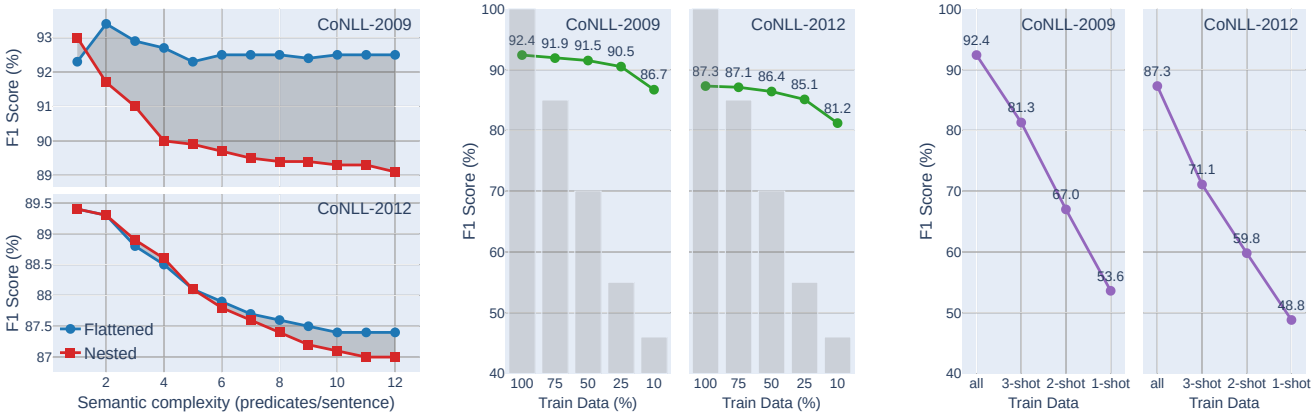
Table 4: Results on the English in-domain test set of the CoNLL-2012 gold benchmark for span-based SRL. P: precision. R: recall.

4.4 Results

Dependency-based SRL. Table 2 summarizes the results on dependency-based SRL in the English in-domain test of CoNLL-2009. Even though GSRL is also tasked to generate predicate sense labels, GSRL$_{\textit{flattened}}$ significantly surpasses the previously best-performing sequence-to-sequence model of Daza and Frank [2019] by 1.6% in F_1 score (17% decrease in error rate).[4] While both systems take advantage of pretrained encoders (BART and ELMo), GSRL also exploits the pretrained decoder of BART, which allows for superior performance. Moreover, when compared to state-of-the-art *sequence labeling* approaches [Shi and Lin, 2019; Conia and Navigli, 2020], GSRL$_{\textit{flattened}}$ shows competitive results, with an F_1 score that is either matching or not statistically different. It is interesting to note that, while GSRL$_{\textit{nested}}$ is tasked to learn semantic structures that can be an order of magnitude more complex than those learnt by its GSRL$_{\textit{flattened}}$ counterpart, the resulting difference in performance is not as large as one may expect, and the training process is more than 60% faster. However, the considerably lower recall shown by GSRL$_{\textit{nested}}$ empirically confirms the complexity of identifying and generating longer sequences of predicate and role labels, especially when a single word is enclosed by multiple labels, i.e., it is an argument for multiple predicates. Furthermore, Table 3 reports the results in the English out-of-domain test of CoNLL-2009 where we observe a similar trend to the in-domain evaluation, with GSRL$_{\textit{flattened}}$ significantly surpassing the previous *sequence-to-sequence* approach [Daza and Frank, 2019] and performing on a par with the state of the art [Conia and Navigli, 2020].

Span-based SRL. Table 4 summarizes the results on span-based SRL in the English test of CoNLL-2012. Similarly to CoNLL-2009, GSRL$_{\textit{flattened}}$ achieves state-of-the-art results in an area where *sequence labeling* approaches are currently predominant. In this setting, however, GSRL$_{\textit{flattened}}$ and GSRL$_{\textit{nested}}$ attain comparable performance, and they both surpass the *sequence-to-sequence* model of Daza and Frank [2018] by a large margin (more than 11.4% in F_1 score). The

[4] Daza and Frank [2019] rely on a separate system trained on a larger amount of sentences in order to output predicate sense labels.



(a) Results as the number of predicates per sentence becomes larger: the gap widens as the semantic complexity increases.

(b) Results of GSRL$_{flattened}$ as the train data decreases: the margin between 100% and 25% is not large.

(c) Results of GSRL$_{flattened}$ as the sentences for each predicate sense decrease: the performance goes down abruptly.

Figure 2: Our analysis shows that i) the semantic complexity of a sentence is the main culprit for the gap in performance between GSRL$_{flattened}$ and GSRL$_{nested}$ (Fig. 2a), but also that ii) GSRL is robust to substantially smaller training datasets (Fig. 2b), and iii) that the number of examples for each predicate sense is fundamental for a good training set (Fig. 2c).

close gap between the two GSRL models can be explained by the lower semantic complexity of the sentences in CoNLL-2012 (see §4.1, Data Statistics), which results in easier SRL structures to be generated. In both span- and dependency-based SRL, it is worth noting that, while the F_1 score of GSRL is on a par with the best-performing *sequence labeling* approaches, GSRL always shows a higher precision.

5 Analysis

In what follows we propose an evaluation framework composed of a set of synthetic scenarios built from the CoNLL-2009 and CoNLL-2012 datasets. Our aim is two-fold: i) to better evaluate the behaviour of GSRL, or any other SRL system, and ii) to gain insights into what is needed for the creation of better training datasets or challenging benchmarks for SRL. In order to enable future comparisons with this work, we release our evaluation framework at `github.com/SapienzaNLP/gsrl`.

Test down-sampling: Semantic complexity. We observe the difference in performance between GSRL$_{flattened}$ and GSRL$_{nested}$ when including increasingly complex sentences in an initially empty test set. To this end, we build 12 test sets from both CoNLL-2009 and CoNLL-2012 by selecting each sentence according to its semantic complexity (see §4.1, Data Statistics), i.e. we collect those sentences containing only 1 predicate, up to 2 predicates, up to 3, and so on. Finally, we evaluate our models on the collected samples. Fig. 2a confirms that the complexity of the semantic structure of a sentence is, indeed, one of the main factors behind the gap between performances of GSRL$_{flattened}$ and GSRL$_{nested}$. This also explains why the two are much closer in CoNLL-2012, as this dataset has a significantly lower semantic complexity than CoNLL-2009 (2.8 against 4.7 predicates per sentence, respectively).

Train down-sampling: Sentence count. Even though unsupervised learning has been gaining ever more popularity in Natural Language Processing, the majority of the approaches to SRL continue to rely on supervision and, therefore, on labeled data. However, the manual annotation of text with sense and role labels is an expensive process which requires money, time and expert annotators who are at ease with complex linguistic resources like PropBank, making it difficult to create large SRL datasets. In this analysis we devise a synthetic scenario in which we simulate a set of lower-resource settings and study how they affect our model. Specifically, we create different training data splits, sampling 10%, 25%, 50% and 75% of the sentences from the training data of CoNLL-2009 and CoNLL-2012 (37,847 and 90,856 sentences, respectively). As shown in Fig. 2b, when down-sampling the training data to 75% and 50% of its original size, the results decrease by less than 1.0% in F_1 score in the test sets of CoNLL-2009 and CoNLL-2012. On one hand, this experiment demonstrates the robustness of our model. On the other hand, it also suggests that the huge effort carried out by the creators of the CoNLL-2012 dataset to manually annotate the last 45,000 sentences of the training set, made our model improve by only 0.9% in F_1 score. In addition, we perform the same experiment with the state-of-the-art sequence labeling system of Conia and Navigli [2020]. The side-by-side comparison is shown in Fig. 3. Despite the drastic architectural difference between two systems, i.e., GSRL being a sequence-to-sequence system as opposed to the sequence labeling approach of Conia and Navigli [2020], and their different behavior in precision and recall, they converge to the same overall performance in terms of F_1 (green line) in each split on both span- and dependency-based evaluations. We argue, therefore, that simply increasing the number of training sentences is not necessarily the best direction towards better datasets and systems.

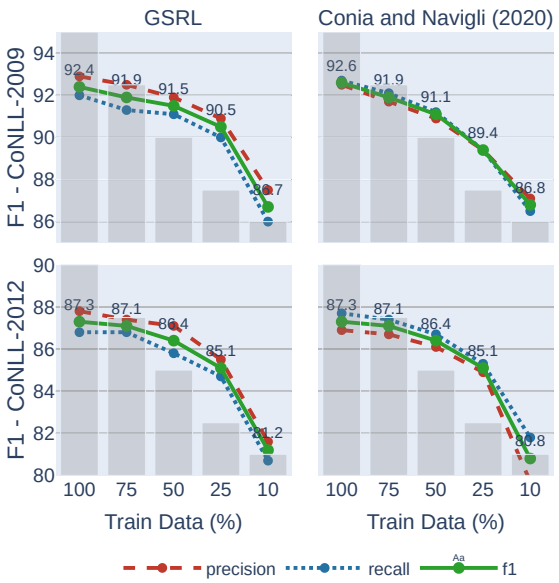


Figure 3: Comparison of GSRL$_{flattened}$ and Conia and Navigli (2020) system results as the train data decreases: the F_1 score is similar in each split (100%, 75%, 50%, 25% and 10% of the original training datasets) of both CoNLL-2009 (dependency-based) and CoNLL-2012 (span-based).

SHOT	CONLL-2009	CONLL-2012
ALL	37,847	90,856
1-SHOT	5,936	4,788
2-SHOT	9,227	8,085
3-SHOT	11,700	10,761

Table 5: Number of sentences in the training samples for 1-, 2- and 3-shot learning.

Train down-sampling: Sense count. Rather than the number of sentences in the training set, we hypothesize that a model is more susceptible to the number of times it sees a predicate sense. To test this hypothesis, we study how well GSRL is able to generalize when limiting the number of sentences for each predicate sense, i.e., how well it performs in few-shot learning. More specifically, we devise a set of three new training datasets which contain at most 1, 2 and 3 occurrences of a predicate sense by sampling the original CoNLL-2009 and CoNLL-2012 training sets. We report the sizes of these new splits in Table 5. Fig. 2c shows the performance of GSRL$_{flattened}$ in both the CoNLL-2009 and CoNLL-2012 test sets as the number of predicate sense instances in the training set decreases. While limiting the number of sentences does not result in a noteworthy impact on the results, GSRL$_{flattened}$ shows a drastic deterioration in performance when it can only learn the predicate-argument structure of a sense from a single example (1-shot), but greatly improves when it can learn from two and three examples (2-shot and 3-shot). Not only do these results support our initial hypothesis, but they also suggest that new smaller-scale datasets, if properly devised, may still make a significant impact on a modern SRL system.

6 Conclusion

In this paper we presented GSRL, the first sequence-to-sequence model for end-to-end SRL to generate both sense and role labels. Evaluated on multiple gold benchmarks, GSRL achieves state-of-the-art results, previously attained only by sequence labeling approaches, in both span- and dependency-based English SRL. The analysis performed on our evaluation framework exposed, thanks to a set of purposely-designed synthetic scenarios, the positives and negatives of our approach, from its ability to reach competitive results with only 25% of the training data to its difficulties in modeling and generating "semantically complex" sequences. However, our analysis was not limited solely to a study of our model and, instead, we also made use of GSRL to highlight current issues, roadblocks and promising directions to further improve the area of SRL, both as regards its models and its datasets. We hope that our contributions will lead to further progress in generation-based approaches to SRL and, more importantly, open the door to their integration into more complex semantics-first tasks, such as Semantic Parsing. We release GSRL and the evaluation framework at `github.com/SapienzaNLP/gsrl`.

Acknowledgments

The authors gratefully acknowledge the support of the ERC Consolidator Grant MOUSSE No. 726487 and the ELEXIS project No. 731015 under the European Union's Horizon 2020 research and innovation programme.

This work was partially supported by the MIUR under the grant "Dipartimenti di eccellenza 2018-2022" of the Department of Computer Science of Sapienza University.

References

[Bahdanau *et al.*, 2015] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proc. of ICLR*, 2015.

[Bevilacqua *et al.*, 2021] Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. One SPRING to Rule Them Both: Symmetric AMR Semantic Parsing and Generation without a Complex Pipeline. *Proc. of AAAI*, 2021.

[Blloshmi *et al.*, 2020] Rexhina Blloshmi, Rocco Tripodi, and Roberto Navigli. XL-AMR: Enabling cross-lingual AMR parsing with transfer learning techniques. In *Proc. of EMNLP*, 2020.

[Cai and Lapata, 2019a] Rui Cai and Mirella Lapata. Semi-supervised Semantic Role Labeling with cross-view training. In *Proc. of EMNLP*, 2019.

[Cai and Lapata, 2019b] Rui Cai and Mirella Lapata. Syntax-aware Semantic Role Labeling without parsing. *Transactions of ACL*, 2019.

[Cai *et al.*, 2018] Jiaxun Cai, Shexia He, Zuchao Li, and Hai Zhao. A full end-to-end semantic role labeler, syntactic-agnostic over syntactic-aware? In *Proc. of COLING*, 2018.

[Carreras and Màrquez, 2005] Xavier Carreras and Lluís Màrquez. Introduction to the CoNLL-2005 shared task: Semantic Role Labeling. In *Proc. of CoNLL*, 2005.

[Caruana, 1997] Rich Caruana. Multitask learning. *Machine Learning*, 1997.

[Chen *et al.*, 2019] Xinchi Chen, Chunchuan Lyu, and Ivan Titov. Capturing argument interaction in Semantic Role Labeling with capsule networks. In *Proc. of EMNLP*, 2019.

[Conia and Navigli, 2020] Simone Conia and Roberto Navigli. Bridging the gap in multilingual Semantic Role Labeling: a language-agnostic approach. In *Proc. of COLING*, 2020.

[Conia *et al.*, 2021] Simone Conia, Andrea Bacciu, and Roberto Navigli. Unifying cross-lingual Semantic Role Labeling with heterogenous linguistic resources. In *Proc. of NAACL*, 2021.

[Daza and Frank, 2018] Angel Daza and Anette Frank. A sequence-to-sequence model for Semantic Role Labeling. In *Proc. of the 3rd Workshop on RepL4NLP*, 2018.

[Daza and Frank, 2019] Angel Daza and Anette Frank. Translate and label! an encoder-decoder approach for cross-lingual semantic role labeling. In *Proc. of EMNLP*, 2019.

[Gildea and Jurafsky, 2002] Daniel Gildea and Daniel Jurafsky. Automatic labeling of semantic roles. *Computational Linguistics*, 2002.

[Gupta and Malik, 2015] Saurabh Gupta and Jitendra Malik. Visual Semantic Role Labeling. *arXiv preprint*, 2015.

[Hajic *et al.*, 2009] Jan Hajic, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Stepánek, Pavel Stranák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proc. of CoNLL*, 2009.

[He *et al.*, 2019] Shexia He, Zuchao Li, and Hai Zhao. Syntax-aware multilingual Semantic Role Labeling. In *Proc. of EMNLP*, 2019.

[Kasai *et al.*, 2019] Jungo Kasai, Dan Friedman, Robert Frank, Dragomir R. Radev, and Owen Rambow. Syntax-aware neural semantic role labeling with supertags. In *Proc. of NAACL*, 2019.

[Lewis *et al.*, 2020] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for Natural Language Generation, Translation, and Comprehension. In *Proc. of ACL*, 2020.

[Li *et al.*, 2019] Zuchao Li, Shexia He, Hai Zhao, Yiqing Zhang, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. Dependency or span, end-to-end uniform Semantic Role Labeling. In *Proc. of AAAI*, 2019.

[Liu *et al.*, 2020] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. In *Proc. of ICLR*, 2020.

[Lyu *et al.*, 2019] Chunchuan Lyu, Shay B. Cohen, and Ivan Titov. Semantic role labeling with iterative structure refinement. In *Proc. of EMNLP*, 2019.

[Marcheggiani and Titov, 2020] Diego Marcheggiani and Ivan Titov. Graph convolutions over constituent trees for syntax-aware Semantic Role Labeling. In *Proc. of EMNLP*, 2020.

[Marcheggiani *et al.*, 2018] Diego Marcheggiani, Jasmijn Bastings, and Ivan Titov. Exploiting semantics in neural Machine Translation with graph convolutional networks. In *Proc. of NAACL*, 2018.

[Màrquez *et al.*, 2008] Lluís Màrquez, Xavier Carreras, Kenneth C. Litkowski, and Suzanne Stevenson. Semantic Role Labeling: An introduction to the special issue. *Computational Linguistics*, 2008.

[Navigli, 2018] Roberto Navigli. Natural Language Understanding: Instructions for (present and future) use. In *Proc. of IJCAI*, 2018.

[Ouchi *et al.*, 2018] Hiroki Ouchi, Hiroyuki Shindo, and Yuji Matsumoto. A span selection model for Semantic Role Labeling. In *Proc. of EMNLP*, 2018.

[Palmer *et al.*, 2005] Martha Palmer, Daniel Gildea, and Paul Kingsbury. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 2005.

[Pradhan *et al.*, 2012] Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Proc. of CoNLL*, 2012.

[Procopio *et al.*, 2021] Luigi Procopio, Rocco Tripodi, and Roberto Navigli. SGL: Speaking the graph languages of semantic parsing via multilingual translation. In *Proc. of NAACL*, 2021.

[Raffel *et al.*, 2020] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text Transformer. *Journal of Machine Learning Research*, 2020.

[Shen and Lapata, 2007] Dan Shen and Mirella Lapata. Using semantic roles to improve Question Answering. In *Proc. of EMNLP*, 2007.

[Shi and Lin, 2019] Peng Shi and Jimmy Lin. Simple BERT models for Relation Extraction and Semantic Role Labeling. *arXiv preprint*, 2019.

[Song *et al.*, 2019] Haoyu Song, Wei-Nan Zhang, Yiming Cui, Dong Wang, and Ting Liu. Exploiting persona information for diverse generation of conversational responses. In *Proc. of IJCAI*, 2019.

[Surdeanu *et al.*, 2008] Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proc. of CoNLL*, 2008.

[Sutskever *et al.*, 2014] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Proc. of NeurIPS*, 2014.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. of NIPS*, 2017.

[Xia *et al.*, 2019] Qingrong Xia, Zhenghua Li, Min Zhang, Meishan Zhang, Guohong Fu, Rui Wang, and Luo Si. Syntax-aware neural semantic role labeling. In *Proc. of AAAI*, 2019.

[Yatskar *et al.*, 2016] Mark Yatskar, Luke S. Zettlemoyer, and Ali Farhadi. Situation recognition: Visual Semantic Role Labeling for image understanding. In *Proc. of CVPR*, 2016.

[Yin *et al.*, 2016] Jun Yin, Xin Jiang, Zhengdong Lu, Lifeng Shang, Hang Li, and Xiaoming Li. Neural generative Question Answering. In *Proc. of IJCAI*, 2016.

[Zhou *et al.*, 2020] Junru Zhou, Zuchao Li, and Hai Zhao. Parsing all: Syntax and semantics, dependencies and spans. In *Findings of EMNLP*, 2020.