Query Your Model with Definitions in FrameNet: An Effective Method for Frame Semantic Role Labeling

Ce Zheng, Yiming Wang, Baobao Chang*

The MOE Key Laboratory of Computational Linguistics, Peking University, China
zcze1112zslx@pku.edu.cn, w10y20ming@gmail.com, chbb@pku.edu.cn

Abstract

Frame Semantic Role Labeling (FSRL) identifies arguments and labels them with frame semantic roles defined in FrameNet. Previous research tends to divide FSRL into argument identification and role classification. Such methods usually model role classification as naive multi-class classification and treat arguments individually, which neglects label semantics and interactions between arguments and thus hindering performance and generalization of models. In this paper, we propose a query-based framework named Argument Extractor with Definitions in FrameNet (AGED) to mitigate these problems. Definitions of frames and frame elements (FEs) in FrameNet can be used to query arguments in text. Encoding text-definition pairs can guide models in learning label semantics and strengthening argument interactions. Experiments show that AGED outperforms previous state-of-the-art by up to 1.3 F1-score in two FrameNet datasets and the generalization power of AGED in zero-shot and few-shot scenarios. Our code and technical appendix is available at https://github.com/PKUnlp-icler/AGED.

Introduction

Semantic Role Labeling (SRL) aims to identify arguments and label them with semantic roles for each predicate in a sentence. Frame Semantic Role Labeling (FSRL) is proposed based on frame semantics theory (Gildea and Jurafsky 2000), where predicates are target words that can evoke semantic frames, and arguments are labeled with frame elements (FEs). Frames and FEs are described in the lexical resource FrameNet (Baker, Fillmore, and Lowe 1998). Frames represent different events, relations, objects, and situations. FEs of a frame are frame-specific roles. FSRL can extract frame semantic structures from text and thus can be helpful to many downstream tasks such as information extraction (Surdeanu et al. 2003), question answering (Shen and Lapata 2007), reading comprehension (Guo et al. 2020). Previous work tends to divide the FSRL into argument identification and role classification, and they usually identify the arguments first and then assign an FE to each argument. There are two flaws in such methods. First, interactions between arguments are either neglected (Kalyanpur et al. 2020; Lin, Sun, and Zhang 2021) or modeled in a sophisticated and time-consuming interaction module (Chen, Zheng, and Chang 2021; Zheng et al. 2022) where the arguments of one frame are identified sequentially. Besides, label semantics of FEs are ignored. For regular role classifiers, representations of FEs are learned sequentially. However, limited by amount of training data of FSRL, such methods may perform poorly, especially for low-resource and few-shot frames because they cannot capture label semantics directly.

In relevant tasks, e.g. SRL (Carreras and Márquez 2005) and Event Argument Extraction (EAE) (Ebner et al. 2020), some researches propose query-based frameworks to handle problems mentioned above. They treat these tasks as Machine Reading Comprehension (MRC) (Liu et al. 2020) or fill-in-the-bank Clozing (Ma et al. 2022). Queries are generated from role-specific questions (Meng et al. 2019) or event (or event role) templates to extract argument spans, where label semantics can be captured in label-specific templates. Recent work in EAE (Ma et al. 2022) uses event templates to extract all arguments simultaneously, where interactions between arguments are highlighted in the event templates. However, the templates or questions used by them are either too simple or in need of elaborative manual design. The simple questions format is not informative enough for FSRL, and manually designed templates cannot be easily applied to FrameNet because there are nearly 1000 frames and 10000 frame elements in FrameNet.

FrameNet describes frames and FEs with frame definitions and FE definitions. As shown in Figure 1, the frame definition of Attack describes an event where an Assailant attacks a Victim, and other FEs of Attack such as Weapon are also included. The definition of Attack also describes how they interact with each other; FE definitions show fine-grained descriptions of FEs and FE relations, for example, the definition of Assailant shows the relation be_attack_by between Victim and Assailant and vice versa.

We find that frame definitions and FE definitions are suitable for the above problems. First, frame definitions describe how FEs of this frame interact with each other, and FE definitions also show fine-grained relations between FEs (e.g. the definition of FE Victim describes relations between Victim and Assailant). We can model relations...
Attack

Definition:
An **Assailant** physically attacks a **Victim** (which is usually but not always sentient), causing or intending to cause the **Victim** physical damage. A **Weapon** used by the **Assailant** may also be mentioned, in addition to the usual **Place**, **Time**, **Purpose**, **Reason**, etc.

<table>
<thead>
<tr>
<th><strong>Term</strong></th>
<th><strong>Definition</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Assailant</strong></td>
<td>The person (or other self-directed entity) that is attempting physical harm to the <strong>Victim</strong>.</td>
</tr>
<tr>
<td><strong>Victim</strong></td>
<td>This FE is the being or entity that is injured by the <strong>Assailant</strong>’s attack.</td>
</tr>
<tr>
<td><strong>Purpose</strong></td>
<td>This FE identifies the purpose for which an <strong>Assailant</strong> attacks the <strong>Victim</strong>.</td>
</tr>
</tbody>
</table>

Is he **INVADING** Iraq just to cover other shortcomings?

Figure 1: Definitions of **Attack** and its FEs **Assailant**, **Victim**, and **Purpose** with an example sentence of frame **Attack** and these FEs. **INVADING** is the target word evoking frame **Attack**, arguments of the target are painted with the same color they correspond to, e.g., *he* is associated with FE **Assailant**.

between FEs with these definitions, and relations between FEs (e.g., **Assailant** and **Victim**) can reflect interactions between arguments (e.g., *he* and *Iraq*). Additionally, we treat definitions as natural language style queries, and representations of FEs are encoded by pretrained language models (PLMs), which can utilize label semantics of FEs, such as FE names and their context in definitions. Moreover, the definitions can be applied directly as templates with only a few modifications, which is informative and free of time-consuming manual design.

In this paper, we propose a query-based frame semantic role labeling framework named **Argument Extractor** with Definitions in FrameNet (AGED). We concatenate the text with frame definitions and use PLMs to encode text-definition pairs. AGED can extract all arguments simultaneously. For each FE, its representation is derived from contextual representations in PLM. Label semantics of this FE, and interactions between FEs can be captured with the bidirectional attention mechanism in PLM. Each FE representation can generate two pointer queries to identify start and end positions of arguments. In addition, we can use text-FE definition pairs as extra training data because FE definitions can represent fine-grained label semantics and FE relations.

Experiments show that AGED outperforms previous state-of-the-art models by up to 1.3 F1-score points in two FrameNet datasets. Further experiments demonstrate the power of AGED in zero-shot and few-shot scenarios. We also combine AGED with Jiang and Riloff (2021) in multitask training paradigm to explore interactions between FSRL and frame identification.

Overall, our contribution can be summarized as follow:

- **We propose AGED**, a query-based framework to model label semantics and strengthen interactions between arguments in FSRL. Different from traditional two-stage (argument identification and role classification) methods, AGED achieve better performance on FSRL, especially in zero-shot and few-shot scenarios.
- **We use definitions in FrameNet as templates in AGED** with only a few modifications. Frame definitions can be used to extract all arguments simultaneously; while FE definitions can serve as additional training data to capture fine-grained label semantics and FE relations.

**Task Formulation**

Frame Semantic Role Labeling aims to identify arguments and label them with frame elements for frame-evoking targets in a sentence. For a sentence $S = w_1, \ldots, w_n$ and a target word $w_t$ that evokes a frame $f$. Suppose that the arguments for the predicate $w_t$ are $a_1, \ldots, a_k$, and we are required to identify the start and end positions $s_i$ and $e_i$ for each argument $a_i = w_{s_i}, \ldots, w_{e_i}$ and label $a_i$ with the semantic role $r_i \in R_f$, where $R_f$ are frame elements of the frame $f$.

Previous researches usually adopt methods including argument identification and role classification:

- **Argument Identification**: the start and end positions $(s_i, e_i)$ for each argument $a_i$ are identified first.
- **Role Classification**: based on $a_i$, $w_t$, and $S$, an FE $r_i \in R_f$ is assigned to $a_i$.

In this work, we use definitions in FrameNet as templates and FEs in definitions as slots; thus, FSRL can be treated as slot filling. Frame definition $D_f$ of frame $f$ contains all FEs $R_f = \{r_1, \ldots, r_m\}$, and we need to fill these slots in $D_f$. For each slot $r_i$:

- First, we need to determine whether there exists an argument $a_i$ labeled with $r_i$ in sentence $S$.
- If there exists an argument $a_i$ labeled with $r_i$ in sentence $S$, we need to identify $s_i$ and $e_i$ for this argument.

**Methodology**

We propose a query-based framework for FSRL named AGED. FSRL is modeled as template clozing. AGED utilize definitions in FrameNet as templates, and FEs in definitions as slots, then we can generate queries for each FE to extract arguments to fill the slots. Figure 2 shows the how AGED build queries from definitions and extract arguments.

**Definitions in FrameNet**

Definitions in FrameNet include frame definitions and FE definitions, which describe label semantics of frames or FEs,
and semantic relations between FEs. AGED encodes text-definition pairs and both definitions can be appended to the text.

**Frame Definitions** Frame definitions describe frames and their FEs in a global view. In other words, the definition of a frame describe how its FEs interact with each other to combine this frame. Frame definitions are similar to event templates in EAE, and according to Ma et al. (2022), using such templates can not only extract all arguments of an event simultaneously, but can also capture argument interactions. If we add frame definitions to the text, AGED can also extract arguments to fill all FE slots in frame definitions and capture argument interactions. However, frames in FrameNet usually contain more semantic roles than events in EAE, while some FEs are not mentioned in frame definitions, especially some non-core FEs. An easy and straightforward solution is to add other FEs not mentioned in the raw definitions to them:

\[ D_f = \text{frame name} | \text{raw def} | \text{FE list} \]  

(1)

For a frame \( f \), we concatenate its name \( \text{frame name} \), its raw definition \( \text{raw def} \) and \( \text{FE list} \) together. For all FEs \( R_f \) of frame \( f \), we remove FEs that have been included in \( \text{raw def} \), and remaining FEs are listed in the order predefined by FrameNet project to construct \( \text{FE list} \). Now \( D_f \) ensures that any role \( r \in R_f \) has a slot in the definition, and AGED can extract all arguments in this frame from text with \( D_f \).

**Frame Element Definitions** Frame element definitions give precise and concrete descriptions of frame elements. FE definitions also show fine-grained relations between FEs and can help AGED capture argument interactions better. For example, the FE definition of Assailant tells the relation \text{be attack by} between Victim and Assailant, and reflects the semantic relation between arguments \text{he} and \text{Iraq}, so it is helpful to capture arguments' interactions to extract arguments. Accordingly, we also append FE definitions to text in training stage as extra training data. For a frame \( f \) and an FE \( r \in R_f \), we concatenate its \text{frame name} and \text{FE name} and \text{raw def} together. Different from frame definition, we use both \text{frame name} and \text{FE name} here to distinguish FEs of different frames that have the same name; we do not add \text{FE list} because for FE definition, we do not aim at extracting all arguments, instead, we only care FEs that are related to current FE.

\[ D_r = \text{frame name} | \text{FE name} | \text{raw def} \]  

(2)

**Definitions v.s. Role Specific Questions** A common approach in query-based framework is to use simple role-specific questions like What is the Assailant of Attack? or Who attacks Iraq? To compare the performance of definitions in FrameNet with simple questions, we also use plain question here:

\[ Q_r = \text{What’s [FE name] of [frame name]}? \]  

(3)

This kind of question template is simple and can be applied to any FE of any frame. However, this template only models label semantics by its name and ignores semantic relations between FEs. Moreover, a text-question pair can only extract one argument, so it will be slower than text-definition pairs.

**Frame definitions v.s. FE definitions** Both the frame definitions and the FE definitions can be appended to the text to extract arguments. Frame definitions are used in both the training and inference stages because frame definitions can be efficient in extracting all arguments from a global view in one shot. FE definitions implies fine-grained label semantics of FEs and semantic relations between FEs.

FE mentions in the definitions are slots to be filled. Slots in frame definitions include all FEs of this frame and slots in FE definitions only include this FE and its related FEs.

**Model Architecture**
AGED, shown in Figure 2, uses PLMs to encode text-definition pairs to contextualized representations and generate queries for each FE slot in the definition. Argument queries can capture label semantics of FEs and semantic relations between arguments, then they are used to identify start and end positions of each arguments.

\(^1\)As is shown in figure 1, there are two Assailant mentions in Attack definitions, and we only use the leftmost mention as its slot.
Text Encoder  Text-definition pairs are fed to the PLMs in the form of [CLS] text [SEP] definition [SEP]. The contextualized representations of each token in both the text and the definition are then derived from PLMs. The definition can be viewed as context of tokens in text and vice versa, thus alignments between arguments in text and FEs in the definition can be learned via self-attention mechanism of PLMs.

\[ H^S; H^D = \text{Encoder} \left( S; D \right) \]  

(4)

\[ H^S = (h_0^S, h_1^S, \ldots, h_n^S) \] are contextualized representations of tokens \( w_1, \ldots, w_n \) in sentence \( S \) and \( h_0^D \) are contextualized representation of token [CLS]. Analogously, \( H^D \) are contextualized representations of tokens in definition \( D \).

To make PLMs focus on targets and frame or FE mentions, we also add special tokens in text-definition pair. \(<t>\) and \(</t>\) are inserted to the left and right of the target word \( w_t \):

\[ S = w_1, \ldots, <t>, w_t, </t>, \ldots, w_n \]  

(5)

Similarly, \(<f>\) and \(</f>\) are placed around frame name, and \(<t>\> \) and \(</t>\> \) are placed around all FE mentions in the definition.

Query Generator and Span Pointer  Based on contextualized representations of tokens in text and definition, AGED generate queries for each FE slot in the definition, and use these queries to identify start and end positions of arguments.

FE slots are FE mentions in the definition. If an FE has more than one mentions in the definition, we choose the leftmost one as its slot. The slots in a frame definition are all FEs of this frame, while slots in an FE definition contain this FE and FEs related to it. For \( m \) slots in the definition \( D \), slot\(_1\), \ldots, slot\(_m\), each slot is an FE mention span \( s_j = w_{j_1}, \ldots, w_{j_d} \), where \( s_j^' \) and \( e_j^' \) are start and end positions of slot\(_j\) in definition \( D \).

\[ q_i = \text{Maxpooling} \left( h_{s_j^{'}}^S, h_{e_j^{'}}^S \right) \]  

(6)

\( q_1, \ldots, q_m \) are query vectors generated from slot\(_1\), \ldots, slot\(_m\). Then we use pointer networks to identify start and end positions of each slot \( i \).

\[ \text{Pr}(s_i|S, D, q_i) = \text{Softmax} \left( (W^s q_i)^\top \cdot H^S \right) \]  

(7)

\[ \text{Pr}(e_i|S, D, q_i) = \text{Softmax} \left( (W^e q_i)^\top \cdot H^S \right) \]  

(8)

Here \( q_i \in \mathbb{R}^d \) and \( H^S = (h_0^S, h_1^S, \ldots, h_n^S) \in \mathbb{R}^{d \times (n+1)} \). \( W^s \) and \( W^e \) are linear transformation matrices in \( \mathbb{R}^{d \times d} \). \( \text{Pr}(s_i|S, D, q_i) \) is the probability distribution of \( (n+1) \) tokens as the start or end position of slot\(_i\) and the token 0 ([CLS]) means no argument.

Training and Inference  In training stage we use both frame definitions and FE definitions. An instance in the training data includes a sentence \( S \), and the target \( w_t \) that invokes the frame \( f \), and the \( k \) arguments \( a_1, \ldots, a_k \) of the target \( t \) labeled with FEs \( r_1, \ldots, r_k \). We use \( (S, D_f) \) as original training data. Furthermore, for each \( r_t \), we use \( (S, D_r) \) as additional training data. The labels of each slot are the start and end positions of arguments related to slots. If there exists an argument \( w_{k_1}, \ldots, w_{k_i} \) labeled with slot\(_t\), the labels of this slot are \( s_i \) and \( e_i \). If there is no argument labeled slot\(_t\), the labels of this slot are \( s_i = 0 \) and \( e_i = 0 \) (that is, we regard [CLS] as no argument). We use cross entropy loss in AGED.

\[ \mathcal{L}_s = \sum_{i=1}^{m} - \log \text{Pr}(s_i|S, D, q_i) \]  

(9)

\[ \mathcal{L}_e = \sum_{i=1}^{m} - \log \text{Pr}(e_i|S, D, q_i) \]  

(10)

\[ \mathcal{L} = 0.5\mathcal{L}_s + 0.5\mathcal{L}_e \]  

(11)

Inference  We only use frame definitions in inference stage because a frame definition contains all FE of this frame and AGED can extract all arguments efficiently in one shot.

For each FE slot, we first assume that there exists an argument, and we adopt a greedy strategy to identify the argument span, which means the probability of an argument span \( (s_i, e_i) \) labeled with \( a_i \) equals \( \Pr(s_i|S, D, q_i) \cdot \Pr(e_i|S, D, q_i) \):

\[ s_i^{pred} = \arg \max_{s_i \neq 0} \Pr(s_i|S, D, q_i) \]  

(12)

\[ e_i^{pred} = \arg \max_{e_i \neq 0} \Pr(e_i|S, D, q_i) \]  

(13)

We also add some constraints:

• To identify a valid span, \( e_i^{pred} \) should be no less than \( s_i^{pred} \).
• If \( \Pr(s_i = 0|S, D, q_i) \cdot \Pr(e_i = 0|S, D, q_i) \), the probability of no argument, is greater than \( \Pr(s_i = s_i^{pred}|S, D, q_i) \cdot \Pr(e_i = e_i^{pred}|S, D, q_i) \), the prediction of this slot will be no argument.

Experiments  We mainly focus on these questions and conduct corresponding experiments:

• What is the performance of AGED in FrameNet datasets?
• What is the difference between definitions and simple role-specific questions?
• Can definitions in FrameNet help AGED capture label semantics? How does AGED perform in few-shot and zero-shot scenario?

Datasets  We evaluate the performance of AGED in two FrameNet datasets: FN 1.5 and FN 1.7. FN 1.7 is an extension version of FN 1.5. Table 1 shows the comparison between these two datasets. The train / dev / test split is the same as Peng et al. (2018). FrameNet also annotates exemplar sentences for frames and their lexical units, and these exemplar sentences are usually used as additional training data in previous researches (Kshirsagar et al. 2015; Yang and Mitchell 2017; Peng et al. 2018; Zheng et al. 2022). We follow these researches and also use exemplar sentences as extra training data.
Table 1: Comparison between FN 1.5 and FN 1.7. FN 1.7 defines more frames and FEs than FN 1.5, and contains more training and test instances. “exem” means exemplar annotated sentences for frames and lexical units in the FrameNet.

<table>
<thead>
<tr>
<th>#frame</th>
<th># FE</th>
<th>#exem</th>
<th>#train / dev / test</th>
</tr>
</thead>
<tbody>
<tr>
<td>FN 1.5</td>
<td>1019</td>
<td>9634</td>
<td>153952 / 17143 / 22333 / 4458</td>
</tr>
<tr>
<td>FN 1.7</td>
<td>1221</td>
<td>11428</td>
<td>192461 / 19875 / 2309 / 6722</td>
</tr>
</tbody>
</table>

Main Results

We compare AGED with previous models for both FN 1.5 and FN 1.7. For fair comparison, we only compare the performance of AGED with models that use PLMs. These models can be roughly divided into two groups: w/o exemplar and w/ exemplar because some models claim that they do not use exemplar instances as additional training data.

Table 2 shows the main results of our experiments. If AGED is trained with only frame definitions, AGED outperforms the previous state of the art by up to 1.2 F1-score (75.6 → 76.84). Using FE definitions as extra training data can help AGED performs better and it will outperform the previous state of the art by up to 1.3 F1-score (75.6 → 76.91). The results show the effectiveness of AGED under all conditions in both datasets, which indicates that the models can benefit from considering label semantics and argument interactions with definitions in FrameNet. Using FE definitions as extra training data can bring up to 0.6 F1-score increase (74.73 → 75.37) because these definitions give concrete and precise descriptions of FEs and reflect fine-grained FE semantic relations.

Discussion

In this section, we conduct further experiments for better understanding of AGED. If not specified, experiments are conducted on FN 1.5 without exemplar sentences because training with exemplar sentences is time-consuming.

Few-shot and Zero-shot Performance Zheng et al. (2022) design a few/zero-shot experiment to validate the transfer learning ability of the Frame Knowledge Graph constructed by them. We follow the same experiment settings to evaluate AGED with three frames Getting, Arriving, Transition_to_state under zero-shot and few-shot scenarios. Table 3 shows the comparison between AGED and KiD under zero-shot and few-shot scenarios. As they do not use PLMs in this experiment, we cannot directly compare zero-shot results of two models (56.25 v.s. 82.45). When removing all instances of these frames, KiD drops much larger than AGED (-14.06 v.s. -3.81), and the situations is the same under few-shot scenarios (full → 32). Definitions can help AGED directly capture label semantics, while KiD uses semantic relations between Frames and FE mappings in semantic-related frames to transfer knowledge. Results show that using definitions directly is much more effective under zero-shot and few-shot scenarios. Even AGED never see instances of these frames in training stage, its performance is still close to AGED with full instances because AGED learns ability to parse definitions as queries in training stage.

As FN 1.7 is an extension version of FN 1.5, we also conduct another zero-shot experiment for AGED. FN 1.7 defines some new frames which are not included in FN 1.5 and we wonder the performance of AGED trained with FN 1.5 on FN 1.7. Table 1 shows that FN 1.7 defines 202 new frames and the test dataset of FN 1.7 includes at least 2264 new instances. Nearly 25% of these 2264 new instances evoke new frames in FN 1.7. Table 4 shows the results of this experiment. As there are some new frames, AGED trained with FN 1.5 performs slightly worse than AGED trained with FN 1.7 (73.72 v.s. 74.73). The narrow gap indicates the zero-shot performance of AGED. To further validate the performance of AGED with unseen frames, we evaluate AGED trained with FN 1.5 on instances of new frames in FN 1.7 test set, and the gap is still narrow (72.86 v.s. 74.73).

The two above experiments both show the generalizability of AGED. Even if these frames are never or seldom seen in training stage, AGED can still perform promising if the corresponding definitions are provided and the model learns how to encode them.

Target Markers and Label Markers A common practice to strengthen contextualized span representations is the use of markers (Baldini Soares et al. 2019; Xiao et al. 2020). Table 5 gives an ablation study of markers in the text-definition pairs. Results show that label markers such as <arg0, arg1, ...> bring negligible improvements because FEs are already natural language labels instead of abstract labels in Propbank (arg0, arg1, ...) and markers seem redundant to capture label semantics of FEs in FrameNet. However, target markers play a vital role in AGED. When we remove target markers, the performance will drop by 10.54 points. Frame semantics is based on predicate-argument semantic structure, and target (predicate) is central in frame semantics. Without target markers, AGED even does not know which word is the target, thus affecting the performance of AGED.

Combination of Frame Definitions and FE Definitions We can use definitions for AGED and we can still use simple questions, what is the difference between simple questions and definitions? Both frame definitions and FE definitions can be used in AGED, why choose frame definitions in the inference stage and why choose FE definitions as additional training data? This section answers these questions.

In this section, we use three baselines:

- Role-Specific QA: we use role specific questions in Eq. 3. A frame that includes m FEs will construct m text-question pairs. Each pair only queries one argument.
- FE Def: using FE Definitions defined in Eq. 2 instead of simple questions. A frame including m FEs will construct m text-definition pairs. Each text-definition pair only queries one argument.
- Frame Def: baseline model in our work, using frame definitions to query all arguments with only one single text-definition pair.

---

Experiments setups and details of these models are listed in https://github.com/PKUnlp-icler/AGED
<table>
<thead>
<tr>
<th>Model</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>semi-CRF (2017)</td>
<td>-</td>
<td>-</td>
<td>73.56</td>
<td>-</td>
<td>-</td>
<td>72.22</td>
</tr>
<tr>
<td>Lin, Sun, and Zhang (2021)</td>
<td>-</td>
<td>-</td>
<td>73.28</td>
<td>-</td>
<td>-</td>
<td>72.06</td>
</tr>
<tr>
<td>Kalyanpur et al. (2020)</td>
<td>71.93</td>
<td>76.78</td>
<td>74.28</td>
<td>74.02</td>
<td>75.46</td>
<td>74.73</td>
</tr>
<tr>
<td>AGED (ours) w/o exemplar</td>
<td>73.43</td>
<td>76.31</td>
<td>74.84</td>
<td>75.39</td>
<td>75.36</td>
<td>75.37</td>
</tr>
<tr>
<td>Chen, Zheng, and Chang (2021)</td>
<td>69.27</td>
<td>75.39</td>
<td>72.20</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Bastianelli, Vanzo, and Lemon (2020)</td>
<td>74.23</td>
<td>76.94</td>
<td>75.56</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>KID (2022)</td>
<td>71.7</td>
<td>79.0</td>
<td>75.2</td>
<td>74.1</td>
<td>77.3</td>
<td>75.6</td>
</tr>
<tr>
<td>AGED (ours)</td>
<td>73.06</td>
<td>79.75</td>
<td>76.30</td>
<td>75.84</td>
<td>77.87</td>
<td>76.84</td>
</tr>
<tr>
<td>AGED (ours) + FE definition</td>
<td>74.04</td>
<td>79.75</td>
<td>76.79</td>
<td>75.80</td>
<td>78.05</td>
<td>76.91</td>
</tr>
</tbody>
</table>

Table 2: Empirical results on the test set of FN 1.5 and FN 1.7. Models in the upper block do not use exemplar instances others in the bottom block use exemplar instances as additional training data. AGED outperforms previous state-of-the-art by up to 1.3 F1 score (75.6 → 76.91). AGED still performs better than other models when trained with only frame definitions.

<table>
<thead>
<tr>
<th>Model</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
<th>∆</th>
</tr>
</thead>
<tbody>
<tr>
<td>KID (GloVe)</td>
<td>56.26</td>
<td>65.95</td>
<td>70.32</td>
<td></td>
</tr>
<tr>
<td>AGED</td>
<td>82.45</td>
<td>84.06</td>
<td>86.26</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Experiments under zero-shot and few-shot scenarios. Experiment settings is from Zheng et al. (2022). In zero-shot scenarios, AGED performs much promising in zero-shot and few-shot scenarios than KID (Zheng et al. 2022). The difference between zero-shot and full instances is quite small (82.45 → 86.26), which verifies that definitions can bring the power of generalization and transfer learning.

<table>
<thead>
<tr>
<th>Model</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
<th>∆</th>
</tr>
</thead>
<tbody>
<tr>
<td>15 → 17</td>
<td>73.28</td>
<td>74.17</td>
<td>73.72</td>
<td>-1.01</td>
</tr>
<tr>
<td>15 → 17 (new)</td>
<td>71.85</td>
<td>73.90</td>
<td>72.86</td>
<td>-1.87</td>
</tr>
<tr>
<td>17 → 17</td>
<td>74.02</td>
<td>75.46</td>
<td>74.73</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 4: Zero-shot experiments from FN 1.5 to FN 1.7. 15 → 17 means training with FN 1.5 and evaluating with FN 1.7. 15 → 17 (new) means evaluating with instances of new frames in FN 1.7. The F1-score of 15 → 17 (new) shows the zero-shot performance of AGED for new frames in FN 1.7.

Results are listed in Table 6. Role-specific simple questions are not informative enough than definitions in FrameNet. Using Frame Def is more efficient than FE Def as a frame includes 10 FEs on average and it will take 10x time to extract arguments when using FE Def. Besides, FE Def gets higher precision, and Frame Def gets higher recall because FE Def gives more detailed description of FEs and Frame Def extracts all arguments in a more global view. A straightforward method to combine strengths of two methods is to use FE Def as extra training data and this method achieves fast and accurate performance.

Multi-Task Learning Frame Identification (FI) and Frame Semantic Role Labeling (FSRL) are both subtasks of Frame Semantic Parsing. Previous research (Bastianelli, Vanzo, and Lemon 2020; Chen, Zheng, and Chang 2021; Lin, Sun, and Zhang 2021; Zheng et al. 2022) has trained their models with these subtasks in an end-to-end framework. Holding the belief that interactions between subtasks can contribute to all subtasks, they usually train their models in multi-task learning. However, as reported in Bastianelli, Vanzo, and Lemon (2020); Lin, Sun, and Zhang (2021), multi-task learning is not beneficial for FSRL which means training with FSRL only performs better than multi-task.

Jiang and Riloff (2021) is similar to AGED because we both feed text-definition pairs to PLMs. We simply reimplement their work by removing lexical unit definitions and model FI as sentence pair multiple choice. We train AGED in both subtasks and also train AGED in multi-task learning. The results are in Table 7. The results of FI are consistent. All frameworks get higher performance than single-task when trained with FI and FSRL. Bastianelli, Vanzo, and Lemon (2020); Lin, Sun, and Zhang (2021) cannot benefit from multi-task learning.
Definitions in FrameNet Definitions in FrameNet are re-used in our work. We use FE definitions as additional training data, because the definitions are not encoded. Different from Jiang and Riloff (2021), AGED concentrates on FSRL, but Jiang and Riloff (2021) and AGED use similar input format. A natural idea is to combine two models in multi-task learning which can explore interactions between these two tasks.

**Query-based Framework** Query-based methods (FitzGerald et al. 2018; Meng et al. 2019; Du and Cardie 2020; Liu et al. 2022) are common in many NLP tasks like Semantic Role Labeling (SRL), Name Entity Recognition (NER) and Event Argument Extraction (EAE). Query-based frameworks generate queries from natural language questions or templates, and these queries are used to extract argument spans from the text that can answer the given questions or fill the slots in template clozing (FitzGerald et al. 2018; Liu et al. 2022). These frameworks show a significant improvement in the generalization of models because label semantics is contained in templates or questions. These templates are either too naive to be informative enough or in need of time-consuming human design. In FSRL, we do not need to worry about this issue because the definitions in FrameNet can directly serve as templates.

Ma et al. (2022) use event templates containing multiple roles and argue that the event templates can be efficient in extracting arguments and aware of argument interactions. Frame definitions in AGED can also extract all arguments in a frame and strengthen relations between them. Besides, we use FE definitions as additional training data, because FE definitions show fine-grained relations between FEs and relations between FEs can reflect argument interactions.

## Conclusion

In this paper, we propose a query-based framework AGED for frame semantic role labeling. Frame definitions and FE definitions can capture label semantics of FEs. Semantic relations between FEs are also included in these definitions. Under the guidance of definitions, AGED achieves fast and accurate performance in two FrameNet datasets. In addition, AGED also shows the strong power of generalization for zero-shot or few-shot frames, which verifies that the label semantics is captured in AGED. Definitions in FrameNet are still potential, and further work can focus on a definition-based end-to-end framework for frame semantic parsing.

### Table 7: AGED trained with single-task vs. multi-task (m).

<table>
<thead>
<tr>
<th>Model</th>
<th>FI Accuracy</th>
<th>Δ</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
<th>Δ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bastianelli, Vanzo, and Lemon (2020)</td>
<td>89.83</td>
<td>-</td>
<td>74.23</td>
<td>76.94</td>
<td>75.56</td>
<td>-</td>
</tr>
<tr>
<td>Bastianelli, Vanzo, and Lemon (2020) (m)</td>
<td>90.10</td>
<td>+0.27</td>
<td>74.56</td>
<td>74.43</td>
<td>74.50</td>
<td>-1.06</td>
</tr>
<tr>
<td>Lin, Sun, and Zhang (2021)</td>
<td>90.16</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>73.56</td>
<td>-</td>
</tr>
<tr>
<td>Lin, Sun, and Zhang (2021) (m)</td>
<td>90.62</td>
<td>+0.46</td>
<td>-</td>
<td>-</td>
<td>73.22</td>
<td>-0.34</td>
</tr>
<tr>
<td>AGED</td>
<td>90.78</td>
<td>-</td>
<td>71.93</td>
<td>76.78</td>
<td>74.28</td>
<td>-</td>
</tr>
<tr>
<td>AGED (m)</td>
<td>91.63</td>
<td>+0.85</td>
<td>72.18</td>
<td>76.89</td>
<td>74.46</td>
<td>+0.18</td>
</tr>
</tbody>
</table>

The results indicate that the use of definitions can narrow the gap between FI and FSRL, because both Jiang and Riloff (2021) and AGED require PLMs to understand the definitions and encode text-definition pairs. Future work on frame semantic parsing can focus on an end-to-end model with definitions.

**Related Work**

**Frame Semantic Role Labeling** Previous researches on FSRL (Kshirsagar et al. 2015; Swayamdipta et al. 2017; Bastianelli, Vanzo, and Lemon 2020; Zheng et al. 2022) range from traditional SVM classifiers to deep neural network architectures like LSTM, GCN, and pretrained language models like BERT. However, they all first find candidate argument spans and then classify them into FEs. Arguments are typically identified by sequence labeling architectures (Swayamdipta et al. 2017; Bastianelli, Vanzo, and Lemon 2020) or span-based methods (Yang and Mitchell 2017; Peng et al. 2018) while interactions between arguments are neglected. Recent researches (Chen, Zheng, and Chang 2021; Zheng et al. 2022) propose well-designed architectures to highlight interactions between arguments by identifying the arguments sequentially. Such methods implicitly consider relations between FEs and are not efficient because of their sequential modeling. Label semantics is also ignored in a typical role classifier, except Zheng et al. (2022), where the ontology frame knowledge graph is used in their work to model structure information between labels.

AGED directly models label semantics and interactions between arguments by encoding text-definition pairs with PLMs so it achieves fast and accurate performance.

**Definitions in FrameNet** Definitions in FrameNet are recently studied and used for Frame Semantic Parsing. Jiang and Riloff (2021); Su et al. (2021) focus on Frame Identification. Jiang and Riloff (2021) traverses all candidate frames for the same target in a sentence and appends lexical unit definition, frame definition to original sentence for each pair, then uses BERT to encode the inputs for further classification. Su et al. (2021) combine frame definition with frame-to-frame relations, and they use frozen BERT to encode frame definition as node features in the frame graph. Besides, Zheng et al. (2022) extract FE relations from FE definitions to construct a frame ontological knowledge graph while the definitions are not encoded. Different from Jiang and Riloff (2021), AGED concentrates on FSRL, but Jiang and Riloff (2021) and AGED use similar input format.

The natural idea is to combine two models in multi-task learning which can explore interactions between these two tasks.
Acknowledgments
This paper is supported by the National Science Foundation of China under Grant No. 61936012 and the National Key Research and Development Program of China under Grant No. 2020AAA0106700.

References


