Meaning Representations for Natural Languages Tutorial Part 3b

Modeling Meaning Representation: AMR

Julia Bonn, **Jeffrey Flanigan**, Jan Hajic, Ishan Jindal, Yunyao Li, Nianwen Xue

Outline

- AMR Parsing
  - Sequence-to-sequence methods
  - Pre/post processing
  - Transition-based methods
  - Graph-based methods
  - Evaluation
- AMR Generation:
  - Sequence-to-sequence methods
  - Graph-based methods
- Silver data
- Pre-training

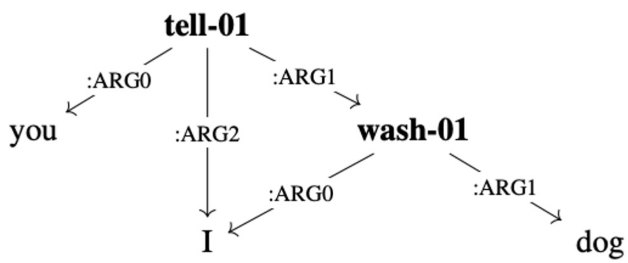
Seq2seq AMR Parsing

- Linearize the AMR graphs
- AMR parsing as sequence-to-sequence modeling
- Can use any seq2seq method and pre-training method (BART, etc)

Konstas et al. Neural AMR: Sequence-to-Sequence Models for Parsing and Generation. ACL 2017.
inter alia.

AMR Linearization

- Linearization order of the AMR graph usually matters



PM (t / tell-01 :ARG0 (y / you) :ARG1 (w / wash-01 :ARG0 i :ARG1 (d / dog)) :ARG2 (i / i))

DFS (<R0> tell-01 :ARG0 (<R1> you) :ARG1 (<R3> wash-01 :ARG0 <R2> :ARG1 (<R4> dog)) :ARG2 (<R2> i))

BFS <R0> tell-01 :ARG0 <R1> you :ARG1 <R3> wash-01 :ARG2 <R2> i <stop> <R3> :ARG0 <R2> :ARG1 <R4> dog <stop>

Figure 1: The AMR graph for the sentence "You told me to wash the dog." with the three different linearizations.

$\text{SPRING}^{\text{DFS}}$	N	<u>83.8</u>
$\text{SPRING}^{\text{BFS}}$	N	83.2
$\text{SPRING}^{\text{PM}}$	N	83.6

Bevilacqua et al. One SPRING to Rule Them Both: Symmetric AMR Semantic Parsing and Generation without a Complex Pipeline. AAAI 2021

AMR Linearization

- Linearization order of the AMR graph usually matters

```
(material
   :mod (raw)
   :domain (opium)
   :ARG1-of (use-01
              :ARG2 (make-01
                       :ARG1 (heroin)
                       :ARG2 (opium))))
```

```
(material
   :domain (opium)
   :mod (raw)
   :ARG1-of (use-01
              :ARG2 (make-01
                       :ARG2 (opium)
                       :ARG1 (heroin))))
```

Figure 5: Example of a variable-free AMR before (left) and after re-ordering (right) for the sentence *Opium is the raw material used to make heroin.*

	Type	Dev	Diff
Baseline	seq2seq	54.8	
AMR Re-ordering	Best	56.8	+ 2.0
	Doubling	60.0	+ 5.2

van Noord & Bos. Neural Semantic Parsing by Character-based Translation: Experiments with Abstract Meaning Representations. Computational Linguistics in the Netherlands Journal. 2017.

Removing Variables

- Remove variables and adding them back-in with post-processing heuristics

```
(m / material
   :mod (r / raw)
   :domain (o / opium)
   :ARG1-of (u / use-01
               :ARG2 (p / make-01
                        :ARG1 (h / heroin)
                        :ARG2 o)))
```

```
(material
   :mod (raw)
   :domain (opium)
   :ARG1-of (use-01
               :ARG2 (make-01
                        :ARG1 (heroin)
                        :ARG2 (opium))))
```

Figure 2: Example of the original AMR (left) and the variable-free AMR (right) displaying the meaning of *Opium is the raw material used to make heroin.*

van Noord & Bos. Neural Semantic Parsing by Character-based Translation: Experiments with Abstract Meaning Representations. Computational Linguistics in the Netherlands Journal. 2017.

Removing Variables

- Rather than removing variables (lossy) use special tokens

```
PM ( t / tell-01 :ARG0 ( y / you ) :ARG1 (
  w / wash-01 :ARG0 i :ARG1 ( d / dog ) )
  :ARG2 ( i / i ) )
DFS ( <R0> tell-01 :ARG0 ( <R1> you ) :ARG1
  ( <R3> wash-01 :ARG0 <R2> :ARG1 ( <R4> dog
  ) ) :ARG2 ( <R2> i ) )
```

Bevilacqua et al. One SPRING to Rule Them Both: Symmetric AMR Semantic Parsing and Generation without a Complex Pipeline. AAAI 2021

Pre-Processing for Transition and Graph-Based: **Recategorization**

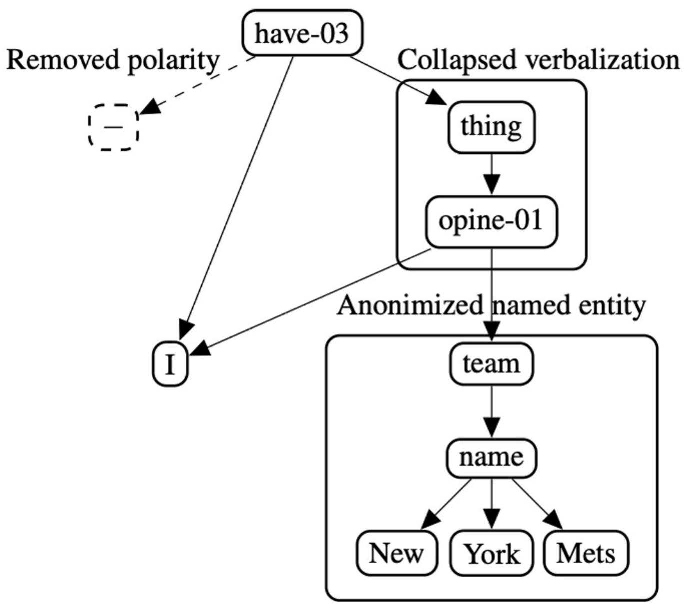


Figure 1: AMR graph of the sentence *I have no opinion on the New York Mets*. Examples of subgraphs for entity anonymization, collapsing of verbalized nouns and removal of the polarity node and edge.

- ❑ Collapsing verbalized concepts
- ❑ Anonymizing named entities (recovered with alignments)
- ❑ Removing sense nodes (predict most frequent sense)
- ❑ Remove wiki links (predict with wikifier)

Zhang et al 2019. AMR Parsing as Sequence-to-Graph Transduction. ACL 2019

Figure from Zhou et al. Structure-aware Fine-tuning of Sequence-to-sequence Transformers for Transition-based AMR Parsing. EMNLP 2021

Outline

- AMR Parsing
  - Sequence-to-sequence methods
  - Pre/post processing
  - **Transition-based methods**
  - Graph-based methods
  - Evaluation

- AMR Generation:
  - Sequence-to-sequence methods
  - Graph-based methods

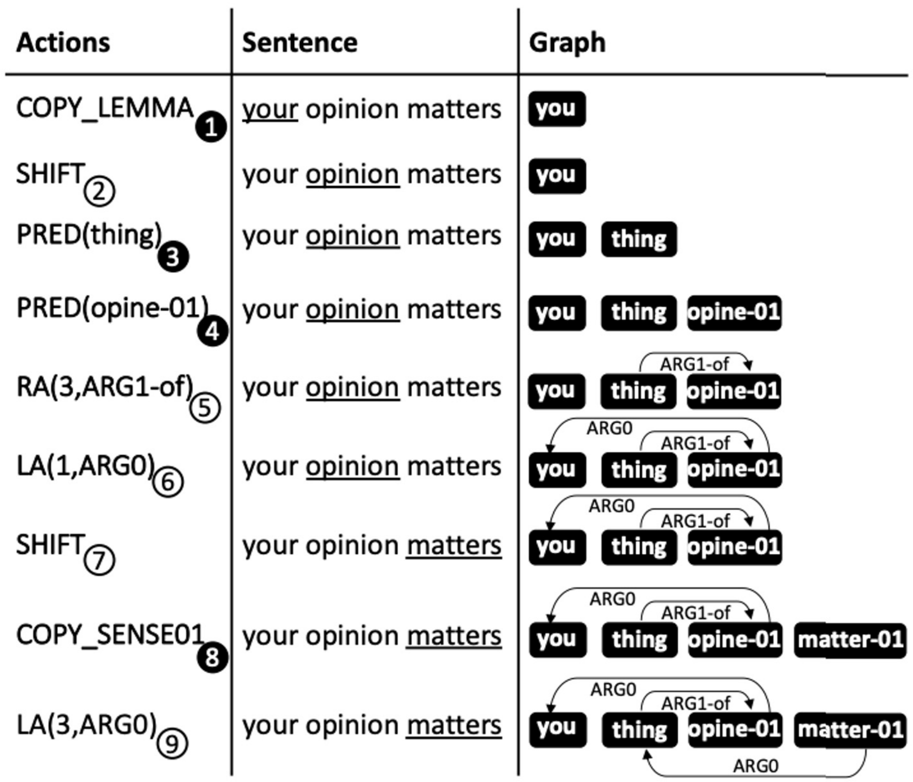
- Silver data
- Pre-training

Transition-Based AMR Parsing

- Construct the graph using a sequence of actions that build the graph
- Use a classifier to predict the next action
- Inspired by transition-based dependency parsing

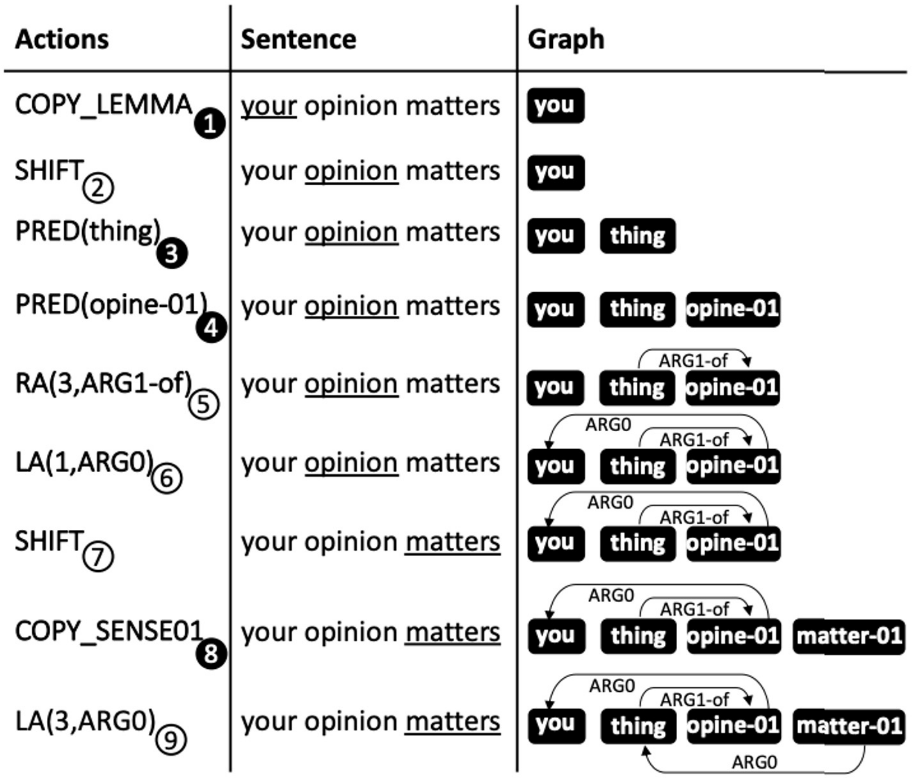
Wang et al. A Transition-based Algorithm for AMR Parsing. NAACL 2015, inter alia.

Transition-Based AMR Parsing



Zhou et al. AMR Parsing with Action-Pointer Transformer.
NAACL 2021

Transition-Based AMR Parsing



Zhou et al. AMR Parsing with Action-Pointer Transformer. NAACL 2021

Simplified Transition Actions

SHIFT moves token cursor one word to the right.

<string> creates node of name <string>.

COPY creates node where the node name is the token under the current cursor position.

LA(j,LBL) creates an arc with label LBL from the last generated node to the node generated at the j_{th} transition step.

RA(j,LBL) same as LA but with arc direction reversed.

ROOT declares the last predicted node as the root.

Zhou et al. Structure-aware Fine-tuning of Sequence-to-sequence Transformers for Transition-based AMR Parsing. EMNLP 2021.

Transition-Based AMR Parsing

- Simplified system: Transition system has 6 actions

SHIFT moves token cursor one word to the right.

<string> creates node of name <string>.

COPY creates node where the node name is the token under the current cursor position.

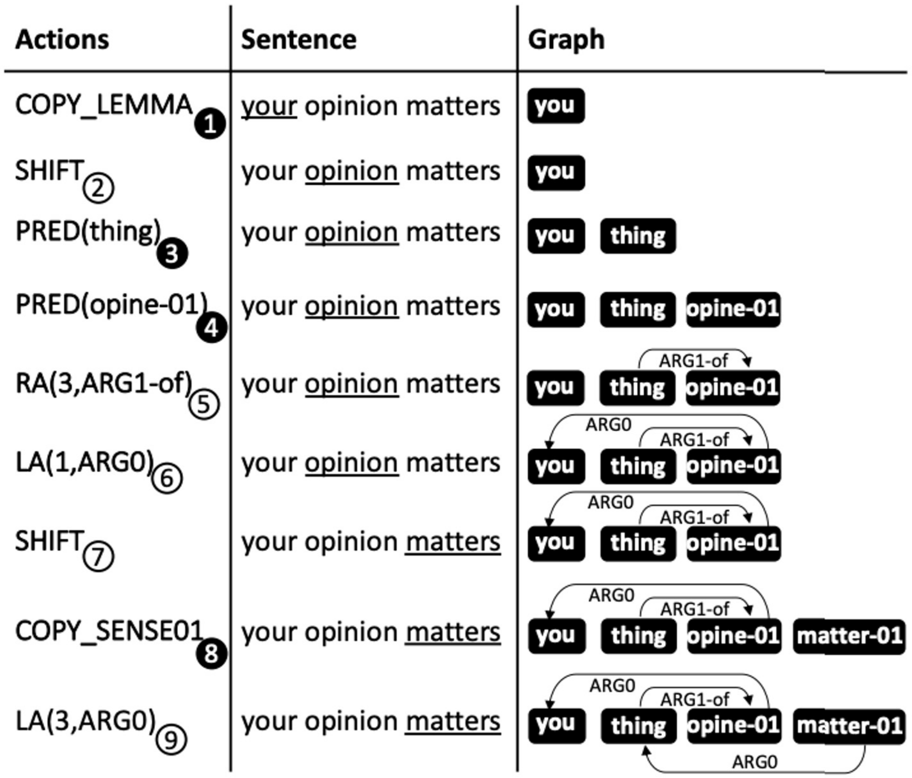
LA(j,LBL) creates an arc with label LBL from the last generated node to the node generated at the j_{th} transition step.

RA(j,LBL) same as LA but with arc direction reversed.

ROOT declares the last predicted node as the root.

Zhou et al. Structure-aware Fine-tuning of Sequence-to-sequence Transformers for Transition-based AMR Parsing. EMNLP 2021.

Transition-Based AMR Parsing



Zhou et al. AMR Parsing with Action-Pointer Transformer. NAACL 2021

Simplified Transition Actions

SHIFT moves token cursor one word to the right.

<string> creates node of name <string>.

COPY creates node where the node name is the token under the current cursor position.

LA(j,LBL) creates an arc with label LBL from the last generated node to the node generated at the j_{th} transition step.

RA(j,LBL) same as LA but with arc direction reversed.

ROOT declares the last predicted node as the root.

Zhou et al. Structure-aware Fine-tuning of Sequence-to-sequence Transformers for Transition-based AMR Parsing. EMNLP 2021.

Transition-Based AMR Parsing

Naseem et al. (2019)	75.5
Cai and Lam (2020)	78.7
Astudillo et al. (2020)	80.2
Bevilacqua et al. (2021)	83.8
Zhou et al. (2021)	81.8
StructBART-S	84.1
StructBART-J	**84.3**

Zhou et al. Structure-aware Fine-tuning of Sequence-to-sequence Transformers for Transition-based AMR Parsing. EMNLP 2021.

Outline

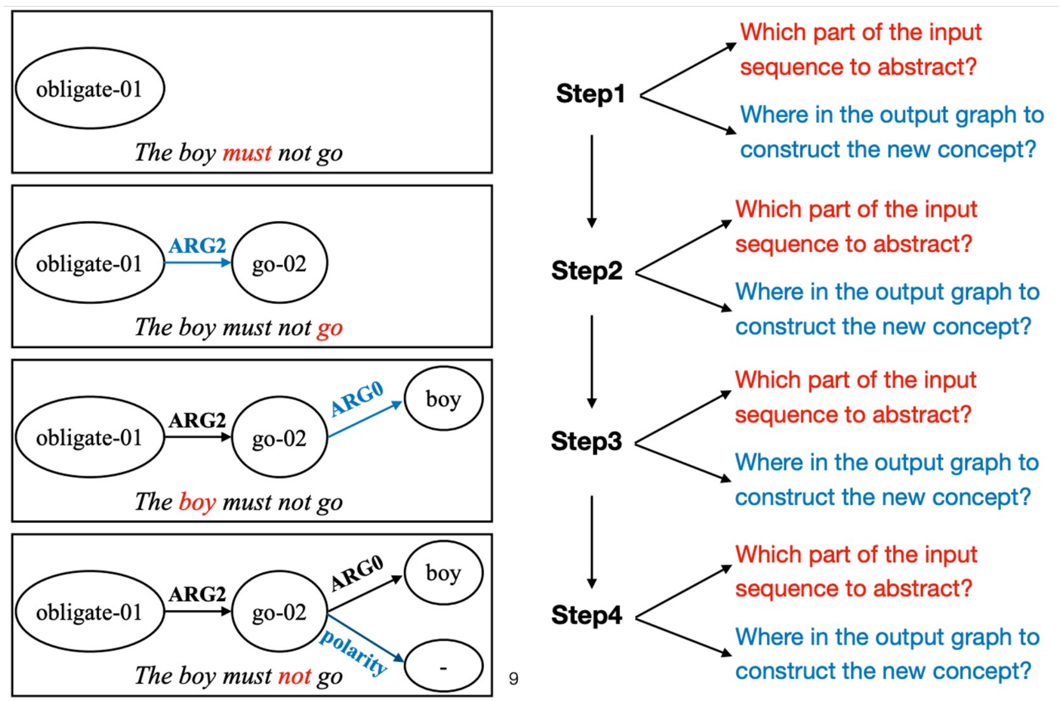
- AMR Parsing
  - Sequence-to-sequence methods
  - Pre/post processing
  - Transition-based methods
  - **Graph-based methods**
  - Evaluation
- AMR Generation:
  - Sequence-to-sequence methods
  - Graph-based methods
- Silver data
- Pre-training

Graph-Based AMR Parsing

- Graph-based methods use the graph structure when predicting
- Inspired by graph-based methods for dependency parsing
- Can be done incrementally or using a structured prediction method

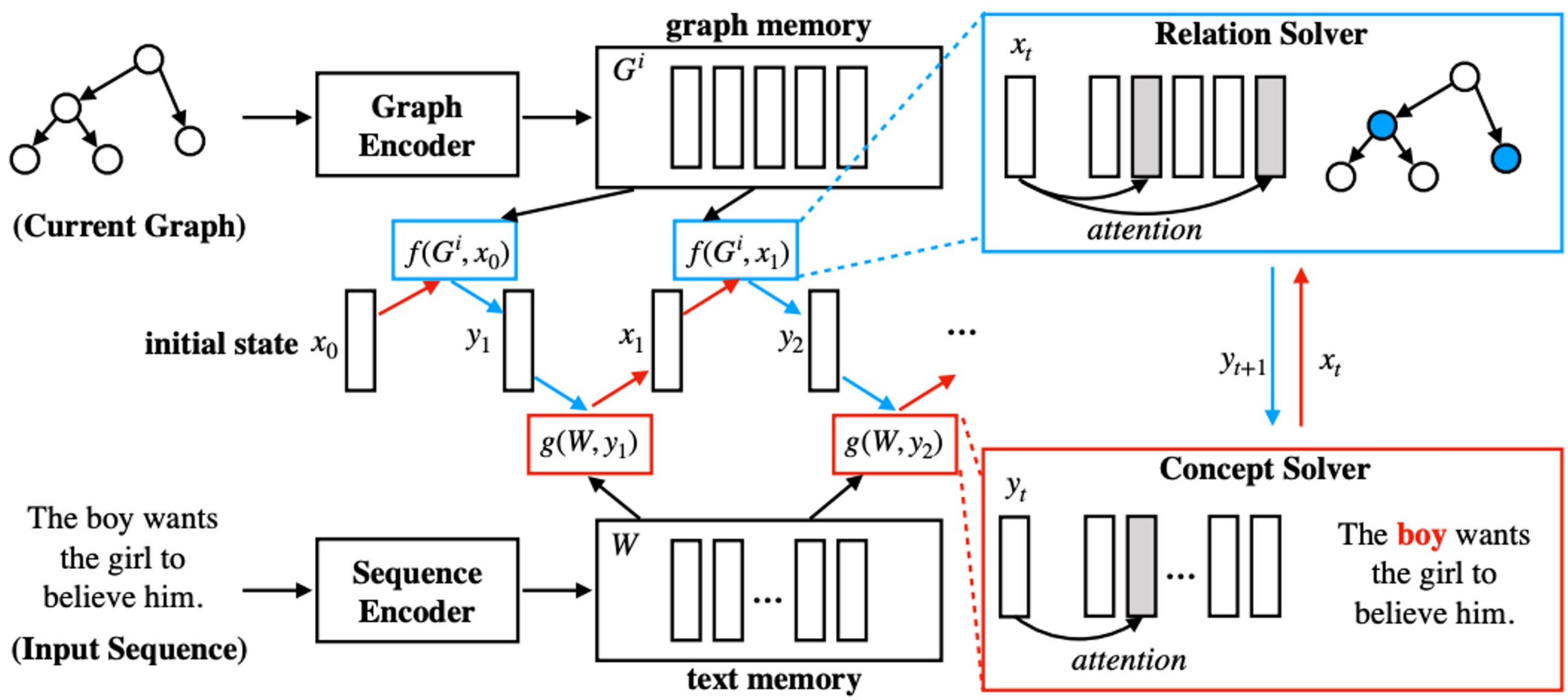
Flanigan et al. A Discriminative Graph-Based Parser for the Abstract Meaning Representation. ACL 2014. inter alia.

Graph-Based AMR Parsing



Cai & Lam. AMR Parsing via Graph-Sequence Iterative Inference. ACL 2020.

Graph-Based AMR Parsing



Cai & Lam. AMR Parsing via Graph-Sequence Iterative Inference. ACL 2020.

Graph-Based AMR Parsing

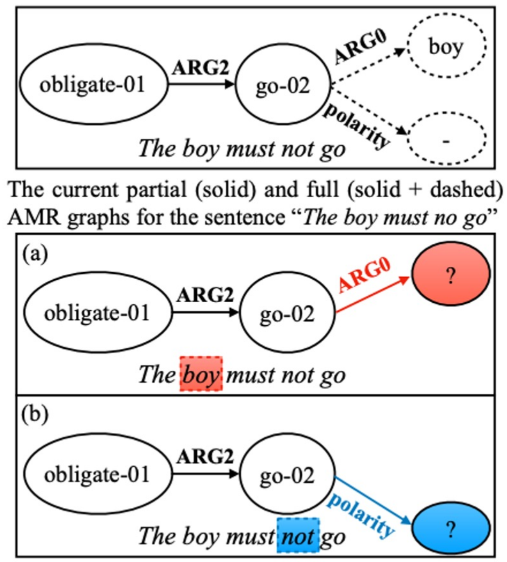


Figure 1: AMR graph construction given the partially constructed graph: (a) one possible expansion resulting in the `boy` concept. (b) another possible expansion resulting in the − (negation) concept.

Cai & Lam 2020. AMR Parsing via Graph-Sequence Iterative Inference. ACL 2020.

Graph-Based AMR Parsing

Model	G. R.	BERT	SMATCH
van Noord and Bos (2017)	×	×	71.0
Groschwitz et al. (2018)	✓	×	71.0
Lyu and Titov (2018)	✓	×	74.4
Cai and Lam (2019)	×	×	73.2
Lindemann et al. (2019)	✓	✓	75.3
Naseem et al. (2019)	✓	✓	75.5
Zhang et al. (2019a)	✓	×	74.6
Zhang et al. (2019a)	✓	✓	76.3
Zhang et al. (2019b)	✓	✓	77.0
Ours	×	×	74.5
	✓	×	77.3
	×	✓	78.7
	✓	✓	**80.2**

Cai & Lam. AMR Parsing via Graph-Sequence Iterative Inference. ACL 2020.

Outline

- AMR Parsing
  - Sequence-to-sequence methods
  - Pre/post processing
  - Transition-based methods
  - Graph-based methods
  - **Evaluation**
- AMR Generation:
  - Sequence-to-sequence methods
  - Graph-based methods
- Silver data
- Pre-training

Evaluation

- Can use fine-grained evaluation to examine strengths and weakness

Metric	First parse	Second parse
Smatch	56	78
Unlabeled	65	100
No WSD	56	78
NP-only	39	86
Reentrancy	69	46
Concepts	56	100
Named Ent.	0	100
Wikification	0	100
Negations	0	0
SRL	69	54

Table 6: Evaluation of the two parses in Figure 5 with the proposed evaluation suite.

Damonte et al. An Incremental Parser for Abstract Meaning Representation. EACL 2017

Outline

- AMR Parsing
  - Sequence-to-sequence methods
  - Pre/post processing
  - Transition-based methods
  - Graph-based methods
  - Evaluation
- **AMR Generation**:
  - Sequence-to-sequence methods
  - Graph-based methods
- Silver data
- Pre-training

AMR Generation: Overview

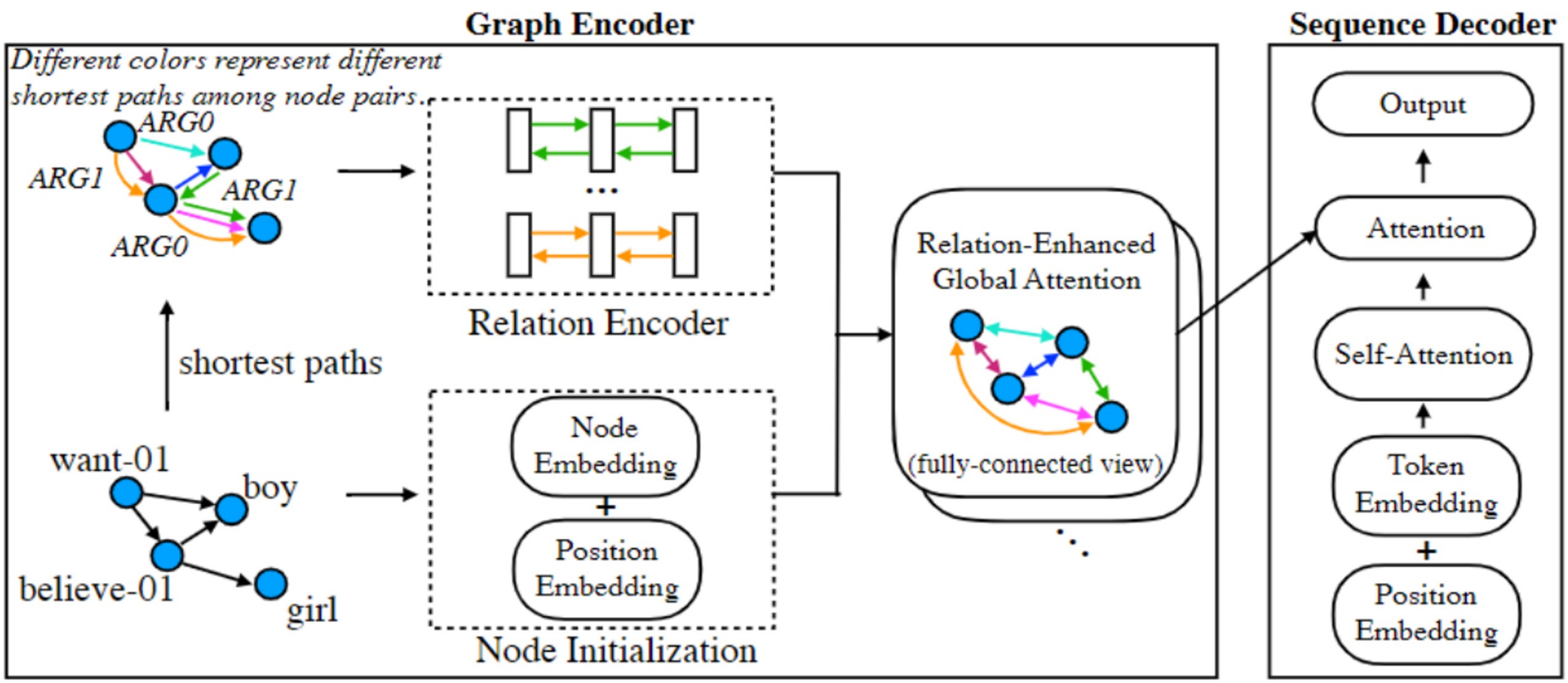
Area	Technique	Paper
Encoder	Rules	Flanigan et al. (2016); Song et al. (2016); Song et al. (2017); Pourdamghani et al. (2016); Manning (2019)
	Seq-to-Seq	Konstas et al. (2017); Cao and Clark (2019); Zhu and Li (2020)
	Graph-to-Seq	Song et al. (2018); Beck et al. (2018); Guo et al. (2019); Zhao et al. (2020); Damonte and Cohen (2019); Ribeiro et al. (2019); Zhang et al. (2020b)
	Transformers	Zhu et al. (2019); Cai and Lam (2020); Wang et al. (2020a); Yao et al. (2020); Jin and Gildea (2020)
	PLM	Mager et al. (2020); Ribeiro et al. (2021a,b); Xu et al. (2021); Bevilacqua et al. (2021); Fan and Gardent (2020)
Other Derivatives	Training Process	Song et al. (2020); Wang et al. (2020b)
	Decoder	Bai et al. (2020)

Hao et al. A Survey : Neural Networks for AMR-to-Text. 2022

AMR Generation: Seq2seq

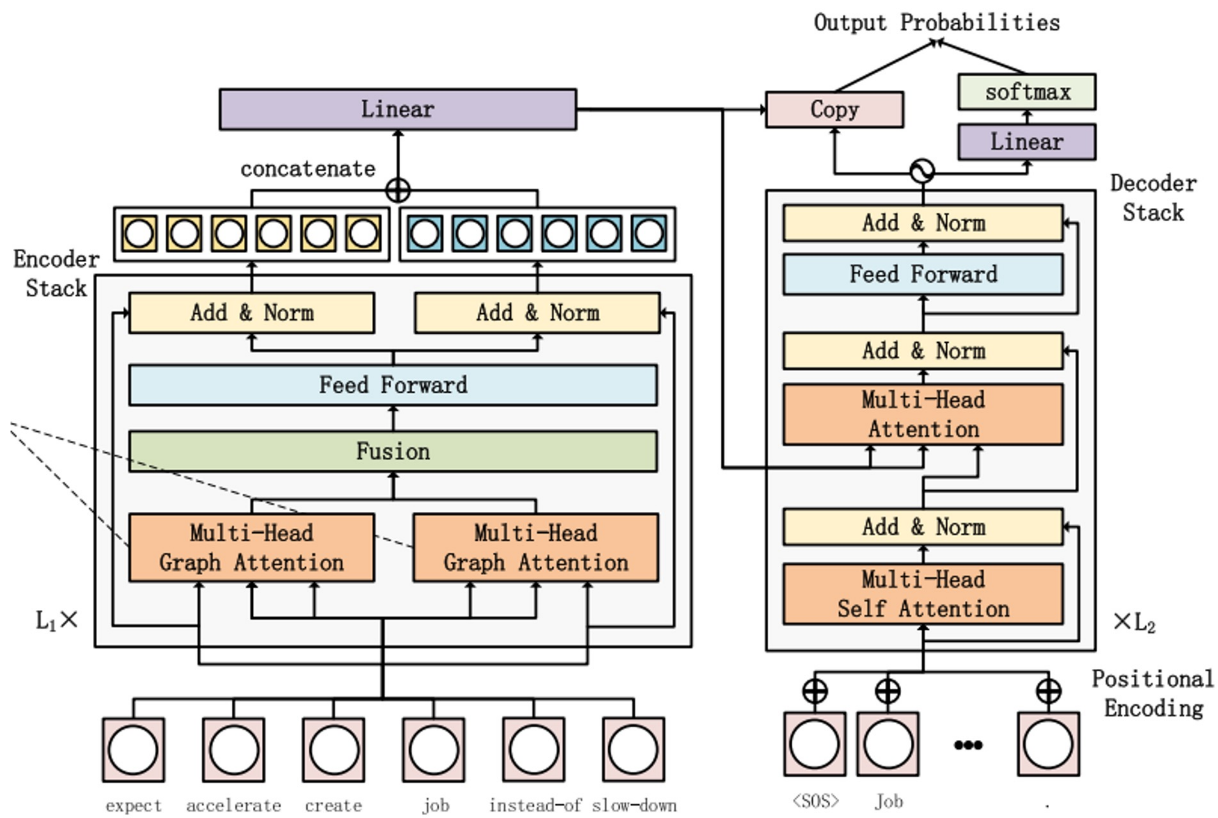
- Linearize the AMR graphs
- AMR generation as sequence-to-sequence modeling
- Can use any seq2seq method and pre-training method (BART, etc)

AMR Generation: Graph-Based



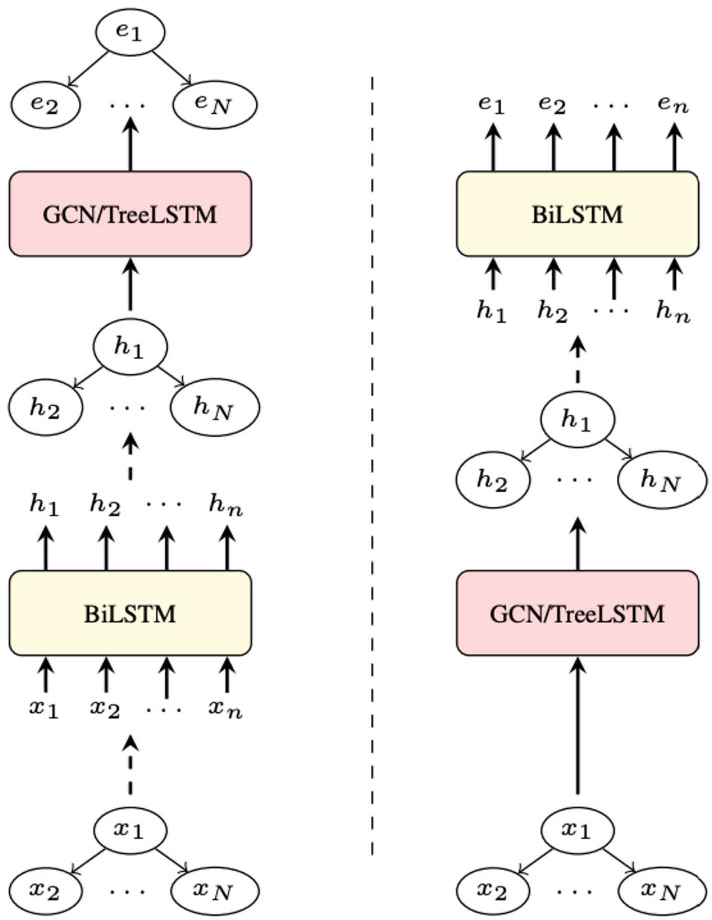
Hao et al. A Survey : Neural Networks for AMR-to-Text. 2022

AMR Generation: Graph-Based



Hao et al. Heterogeneous Graph Transformer for Graph-to-Sequence Learning. ACL 2020

AMR Generation: Graph-Based



Damonte & Cohen. Structural Neural Encoders for AMR-to-text Generation. NAACL 2019

AMR Generation: Comparison

Models	LDC2015E86		LDC2017T10	
	BELU	Meteor	BELU	Meteor
Sequence-Based Model				
Seq2Seq (Konstas et al. 2017)	22.0	-	-	-
Seq2Seq + Syntax (Cao and Clark 2019)	23.5	-	26.8	-
Seq2Seq + SA-based (Zhu and Li 2020)	29.66	35.4	31.54	36.02
Seq2Seq + CNN-based (Zhu and Li 2020)	29.1	35.0	31.82	36.38
Graph-Based Model				
Graph2Seq+CharLSTM+Copy (Song et al. 2018)	22.8	-	-	-
Graph2Seq (Beck et al. 2018)	27.5	-	-	-
GCNSEQ (Damonte and Cohen 2019)	24.4	23.6	24.5	24.0
Dual Graph (Ribeiro et al. 2019)	24.3	30.5	27.8	33.2
LDGCN-GC (Zhang et al. 2020b)	30.8	36.4	33.6	37.5
Line Graph + MixGAT (Zhao et al. 2020)	30.6	35.8	32.5	36.8
Transformer-Based Model				
Transformer (Zhu et al. 2019)	25.5	33.2	27.4	34.6
Graph Transformer (Wang et al. 2020a)	25.9	-	29.3	-
GTransformer (Cai and Lam 2020)	27.4	32.9	29.8	35.1
ADJMATMUL (Jin and Gildea 2020)	-	-	31.2	-
HetGT (Yao et al. 2020)	31.8	36.9	34.1	38.1
PLM-Based Model				
GPT-2L Rec.(Mager et al. 2020)	-	-	32.47	36.8
T5-Large (Ribeiro et al. 2021a)	-	-	45.8	43.85
T5-Large STRUCTADAPT (Ribeiro et al. 2021b)	-	-	46.62	-
SPRING (Bevilacqua et al. 2021)	-	-	45.9	41.8

Hao et al. A Survey : Neural Networks for AMR-to-Text. 2022

Outline

- AMR Parsing
  - Sequence-to-sequence methods
  - Pre/post processing
  - Transition-based methods
  - Graph-based methods
  - Evaluation

- AMR Generation:
  - Sequence-to-sequence methods
  - Graph-based methods

- **Silver data**
- Pre-training

Silver Data (Semi-supervised learning)

- **Gold data** is human labeled data
- **Silver data** is where you run an existing parser on unlabeled data
- You can add silver data to the training data to improve performance
- Usually people use Gigaword for the silver data (more on this later)

Silver Data for AMR Parsing

- Silver data sometimes helps parsing, usually on out-of-domain data

In-domain

	CaiL	CaiL+r	S^{DFS}	S^{DFS}+s	S^{DFS}+r
Text-to-AMR					
Smatch	78.0	76.7	**83.0**	**83.0**	80.2

Out-of-domain

	New3	TLP	Bio
Text-to-AMR			
SPRING^{DFS} (ID)	78.6	-	79.9
SPRING^{DFS}	**73.7**	77.3	**59.7**
SPRING^{DFS}+recat	63.8	76.2	49.5
SPRING^{DFS}+silver	71.8	**77.5**	59.5

Bevilacqua et al. One SPRING to Rule Them Both: Symmetric AMR Semantic Parsing and Generation without a Complex Pipeline. AAAI 2021

Silver Data for AMR Generation

- Silver data always helps generation, but be careful! **Results are misleading!**

In-domain (official test sets)

Baseline	+Silver data
44.9	**46.5**

- Silver data hurts out of domain data

Out-of-domain

	New3	TLP	Bio
AMR-to-Text			
SPRING^{DFS} (ID)	61.5	-	32.3
SPRING^{DFS}	**51.7**	**41.5**	5.2
SPRING^{DFS}+silver	50.2	40.4	**5.9**

Bevilacqua et al. One SPRING to Rule Them Both: Symmetric AMR Semantic Parsing and Generation without a Complex Pipeline. AAAI 2021

Silver Data for AMR Generation

- Silver data always helps generation, but be careful! **Results are misleading!**

Du & Flanigan. Avoiding Overlap in Data Augmentation for AMR-to-Text Generation. ACL 2020

Silver Data for AMR Generation

- Recommend excluding parts of Gigaword that may overlap with test data

	No Extra Data	Baseline Strategy	no-ID	no-Month	no-3Months
Overall	27.58	34.46	33.53	33.44	33.16
Bolt	17.36	21.37	21.20	22.66	19.7
Consensus	20.18	25.96	27.18	26.44	25.06
Dfa	21.45	24.78	22.81	24.79	23.61
Proxy	31.56	39.81	38.84	38.09	38.39
Xinhua	25.22	32.59	31.68	31.77	32.40

https://github.com/jlab-nlp/amr-clean

Du & Flanigan. Avoiding Overlap in Data Augmentation for AMR-to-Text Generation. ACL 2020

Outline

- AMR Parsing
  - Sequence-to-sequence methods
  - Pre/post processing
  - Transition-based methods
  - Graph-based methods
  - Evaluation
- AMR Generation:
  - Sequence-to-sequence methods
  - Graph-based methods
- Silver data
- **Pre-training**

AMR Parsing: Pretraining

- Pre-training the encoder, such as BERT, helps a lot
- Pre-training the decoder, such as BART, helps even more
- Structural pre-training helps as well

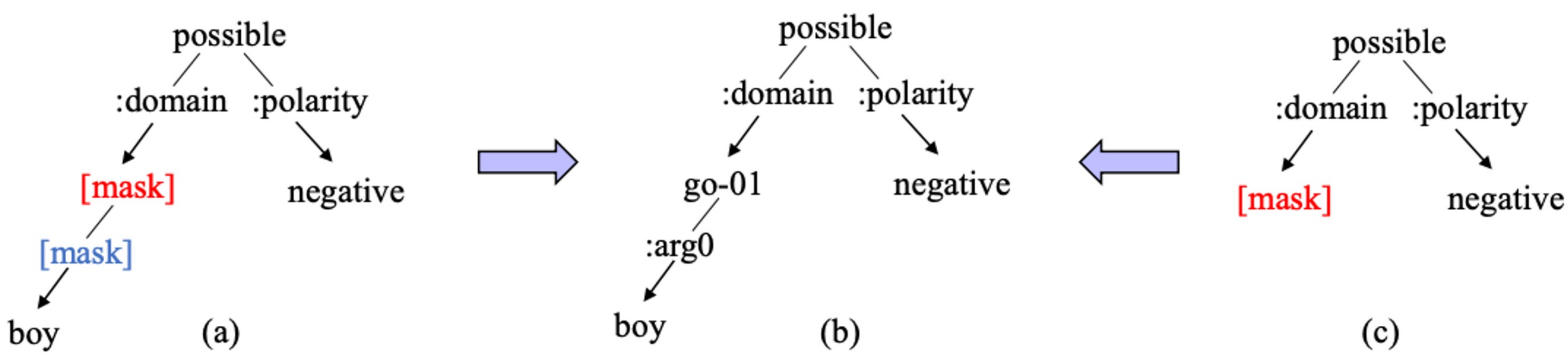


Figure 2: Illustration of two graph pre-training strategies: 1) node/edge level denoising (a→ b); 2) sub-graph level denoising (c→ b). Two transformations can be composed.

Bai et al. Graph Pre-training for AMR Parsing and Generation. ACL 2022

Structural Pretraining

- Structural pre-training helps as well

Task	Input	Output
$\hat{t}$2t	<s> $x_1, ..$ [mask] $.., x_n$ </s>	<s> $x_1, x_2, ..., x_n$ </s>
$\hat{g}$2g	<g> $g_1, ..$ [mask] $.., g_m$ </g>	<g> $g_1, g_2, ..., g_m$ </g>
g2t	<g> $g_1, g_2, ..., g_m$ </g>	<s> $x_1, x_2, ..., x_n$ </s>
t2g	<s> $x_1, x_2, ..., x_n$ </s>	<g> $g_1, g_2, ..., g_m$ </g>
$\hat{t}\bar{g}$2t	<s> $x_1, ..$ [mask] $.., x_n$ </s> <g> [mask] </g>	<s> $x_1, x_2, ..., x_n$ </s>
$\bar{t}\hat{g}$2g	<s> [mask] </s><g> $g_1, ..$ [mask] $.., g_m$ </g>	<g> $g_1, g_2, ..., g_m$ </g>
$\hat{t}$g2t	<s> $x_1, ..$ [mask] $.., x_n$ </s> <g> $g_1, g_2, ..., g_m$ </g>	<s> $x_1, x_2, ..., x_n$ </s>
t$\hat{g}$2g	<s> $x_1, x_2, ..., x_n$ </s> <g> $g_1, ..$ [mask] $.., g_m$ </g>	<g> $g_1, g_2, ..., g_m$ </g>
$\hat{t}\hat{g}$2t	<s> $x_1, ..$ [mask] $.., x_n$ </s> <g> $g_1, ..$ [mask] $.., g_m$ </g>	<s> $x_1, x_2, ..., x_n$ </s>
$\hat{t}\hat{g}$2g	<s> $x_1, ..$ [mask] $.., x_n$ </s> <g> $g_1, ..$ [mask] $.., g_m$ </g>	<g> $g_1, g_2, ..., g_m$ </g>
$\bar{t}$g2t	<s> [mask] </s> <g> $g_1, g_2, ..., g_m$ </g>	<s> $x_1, x_2, ..., x_n$ </s>
t$\bar{g}$2g	<s> $x_1, x_2, ..., x_n$ </s> <g> [mask] </g>	<g> $g_1, g_2, ..., g_m$ </g>

Bai et al. Graph Pre-training for AMR Parsing and Generation. ACL 2022

Structural Pretraining

- Structural pre-training helps as well

Task	Input	
t̂2t	<s> $x_1, ..$ [mask] $.., x_n$ </s>	
ĝ2g	<g> $g_1, ..$ [mask] $.., g_m$ </g>	
g2t	<g> $g_1, g_2, ..., g_m$ </g>	
t2g	<s> $x_1, x_2, ..., x_n$ </s>	
t̂ḡ2t	<s> $x_1, ..$ [mask] $.., x_n$ </s> <g> [mask] </g>	
t̄ĝ2g	<s> [mask] </s><g> $g_1, ..$ [mask] $.., g_m$ </g>	
t̂g2t	<s> $x_1, ..$ [mask] $.., x_n$ </s> <g> $g_1, g_2, ..., g_m$ </g>	
tĝ2g	<s> $x_1, x_2, ..., x_n$ </s> <g> $g_1, ..$ [mask] $.., g_m$ </g>	
t̂ĝ2t	<s> $x_1, ..$ [mask] $.., x_n$ </s> <g> $g_1, ..$ [mask] $.., g_m$ </g	
t̂ĝ2g	<s> $x_1, ..$ [mask] $.., x_n$ </s> <g> $g_1, ..$ [mask] $.., g_m$ </g	
tg2t	<s> [mask] </s> <g> $g_1, g_2, ..., g_m$ </g>	<s> $x_1, x_2, ..., x_n$ </s>
tḡ2g	<s> $x_1, x_2, ..., x_n$ </s> <g> [mask] </g>	<g> $g_1, g_2, ..., g_m$ </g>

Setting	Smatch	BLEU
BART-base	82.7	42.5
+ t̂ḡ2t	82.9	42.9
+ t̄ĝ2g	83.1	42.6
+ t̂ḡ2t, t̄ĝ2g	83.1	42.8
+ t̂ḡ2t, t̄ĝ2g, tĝ2g	83.4	42.8
+ t̂ḡ2t, t̄ĝ2g, t̂g2t	83.1	45.3
+ t̂ḡ2t, t̄ĝ2g, tĝ2g, t̂g2t	83.3	45.0
+ t̂ḡ2t, t̄ĝ2g, t̂ĝ2g	83.2	43.0
+ t̂ḡ2t, t̄ĝ2g, t̂ĝ2t	83.1	44.2
+ t̂ḡ2t, t̄ĝ2g, t̂ĝ2g, t̂ĝ2t	83.2	44.0
+ ALL	83.6	45.6

Bai et al. Graph Pre-training for AMR Parsing and Generation. ACL 2022

AMR Generation: Pretraining

Models	LDC2015E86		LDC2017T10	
	BELU	Meteor	BELU	Meteor
Sequence-Based Model				
Seq2Seq (Konstas et al. 2017)	22.0	-	-	-
Seq2Seq + Syntax (Cao and Clark 2019)	23.5	-	26.8	-
Seq2Seq + SA-based (Zhu and Li 2020)	29.66	35.4	31.54	36.02
Seq2Seq + CNN-based (Zhu and Li 2020)	29.1	35.0	31.82	36.38
Graph-Based Model				
Graph2Seq+CharLSTM+Copy (Song et al. 2018)	22.8	-	-	-
Graph2Seq (Beck et al. 2018)	27.5	-	-	-
GCNSEQ (Damonte and Cohen 2019)	24.4	23.6	24.5	24.0
Dual Graph (Ribeiro et al. 2019)	24.3	30.5	27.8	33.2
LDGCN-GC (Zhang et al. 2020b)	30.8	36.4	33.6	37.5
Line Graph + MixGAT (Zhao et al. 2020)	30.6	35.8	32.5	36.8
Transformer-Based Model				
Transformer (Zhu et al. 2019)	25.5	33.2	27.4	34.6
Graph Transformer (Wang et al. 2020a)	25.9	-	29.3	-
GTransformer (Cai and Lam 2020)	27.4	32.9	29.8	35.1
ADJMATMUL (Jin and Gildea 2020)	-	-	31.2	-
HetGT (Yao et al. 2020)	31.8	36.9	34.1	38.1
PLM-Based Model				
GPT-2L Rec.(Mager et al. 2020)	-	-	32.47	36.8
T5-Large (Ribeiro et al. 2021a)	-	-	45.8	43.85
T5-Large STRUCTADAPT (Ribeiro et al. 2021b)	-	-	46.62	-
SPRING (Bevilacqua et al. 2021)	-	-	45.9	41.8

- **Pre-training helps a lot**
- **Pre-training the encoder and decoder helps the most (BART)**

Hao et al. A Survey : Neural Networks for AMR-to-Text. 2022

AMR Generation: Pretraining

Models	LDC2015E86		LDC2017T10	
	BELU	Meteor	BELU	Meteor
Sequence-Based Model				
Seq2Seq (Konstas et al. 2017)	22.0	-	-	-
Seq2Seq + Syntax (Cao and Clark 2019)	23.5	-	26.8	-
Seq2Seq + SA-based (Zhu and Li 2020)	29.66	35.4	31.54	36.02
Seq2Seq + CNN-based (Zhu and Li 2020)	29.1	35.0	31.82	36.38
GTransformer (Cai and Lam 2020)	27.4	32.9	29.8	35.1
ADJMATMUL (Jin and Gildea 2020)	-	-	31.2	-
HetGT (Yao et al. 2020)	31.8	36.9	34.1	38.1
PLM-Based Model				
GPT-2L Rec.(Mager et al. 2020)	-	-	32.47	36.8
T5-Large (Ribeiro et al. 2021a)	-	-	45.8	43.85
T5-Large STRUCTADAPT (Ribeiro et al. 2021b)	-	-	46.62	-
SPRING (Bevilacqua et al. 2021)	-	-	45.9	41.8

PLM-Based Model				
GPT-2L Rec.(Mager et al. 2020)	-	-	32.47	36.8
T5-Large (Ribeiro et al. 2021a)	-	-	45.8	43.85
T5-Large STRUCTADAPT (Ribeiro et al. 2021b)	-	-	46.62	-
SPRING (Bevilacqua et al. 2021)	-	-	45.9	41.8

- ❑ **Pre-training helps a lot**
- ❑ **Pre-training the encoder and decoder helps the most (BART)**

Hao et al. A Survey : Neural Networks for AMR-to-Text. 2022

Lots More Work

- There's a lot more work we didn't have time to cover
- See the AMR bibliography

https://nert-nlp.github.io/AMR-Bibliography/