Tag Embedding and Well-defined Intermediate Representation improve Auto-Formulation of Problem Description

[NeurIPS 2022 NL4Opt Competition (Subtask 2) 2nd Place]

POSTECH Data Intelligence Lab

Sanghwan Jang

POSTECH

Auto-Formulation of Problem Description

- Auto-formulation is the task of converting an optimization problem into a canonical representation.

Tag: VAR, PARAM, CONST_DIR, LIMIT, OBJ_DIR, OBJ_NAME

A hotel employs cleaners and receptionists. Cleaners earn $500 per week and receptionists earn $350 per week. The hotel requires a minimum of 100 workers of whom at least 20 must be receptionists. To keep the hotel clean and running smoothly, the number of receptionists should be at least a third of the number of cleaners. The hotel wants to keep the weekly wage bill below $30000. Formulate a LP to minimize the wage bill.

Problem Description

Auto-Formulation

500	350	0	0	
0	-1	0	0	-20
0.3333	-1	0	0	0
-1	-1	0	0	-100
500	350	0	0	30000

Canonical Representation

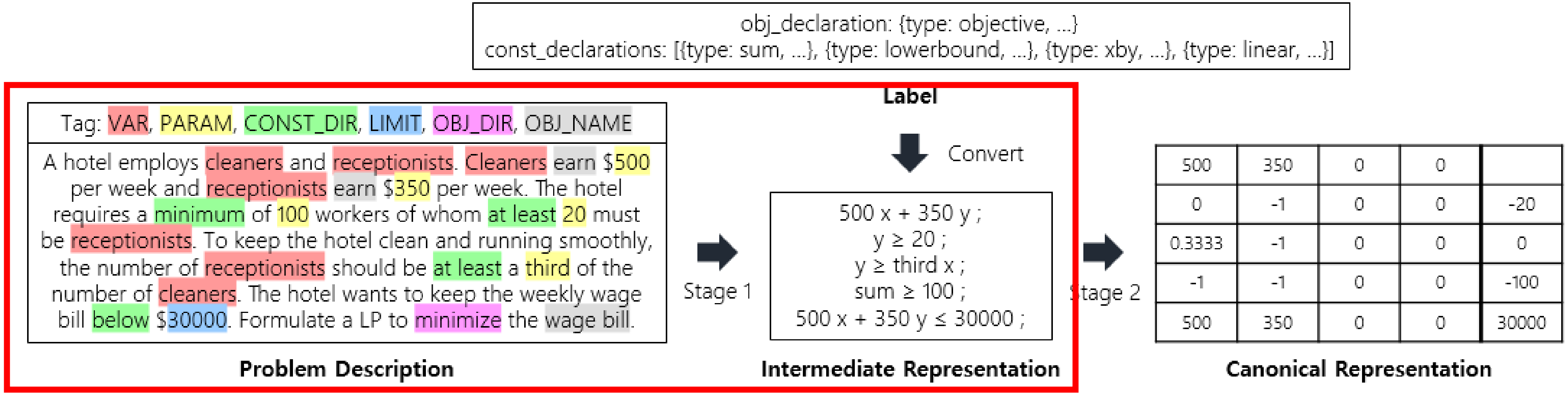
Optimization Solver

- A problem description and tagged entities are given, and we should extract the objective and constraints.

POSTECH

Two-Stage Auto-Formulation

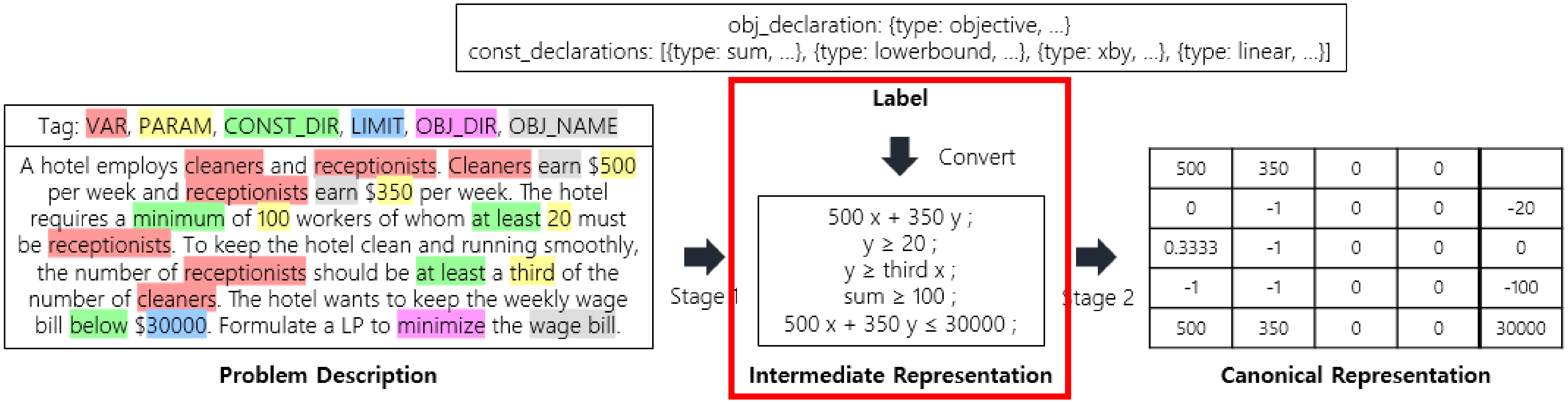
- We decompose auto-formulation task into two stages.
  - (1) optimization problem → intermediate representation
  - (2) Intermediate representation → canonical representation



- The model only needs to focus on the first stage.

Intermediate Representation

- We define intermediate representation to be in the form of mathematical expression.



- The objective and constraints are generated at once to include the relationship between declarations in training.
- To avoid inconsistency in model training, we define the order of generation of declarations.
  (1) objective → constraints (lowerbound → upperbound → xy → xby → sum → linear → ratio)
  (2) position
  (3) x → y → z → w
  (4) ≤ → ≥

POSTECH

Data Augmentation

- We augment the data by reversing the direction of some constraints.

  (1) must not, can not, cannot → must

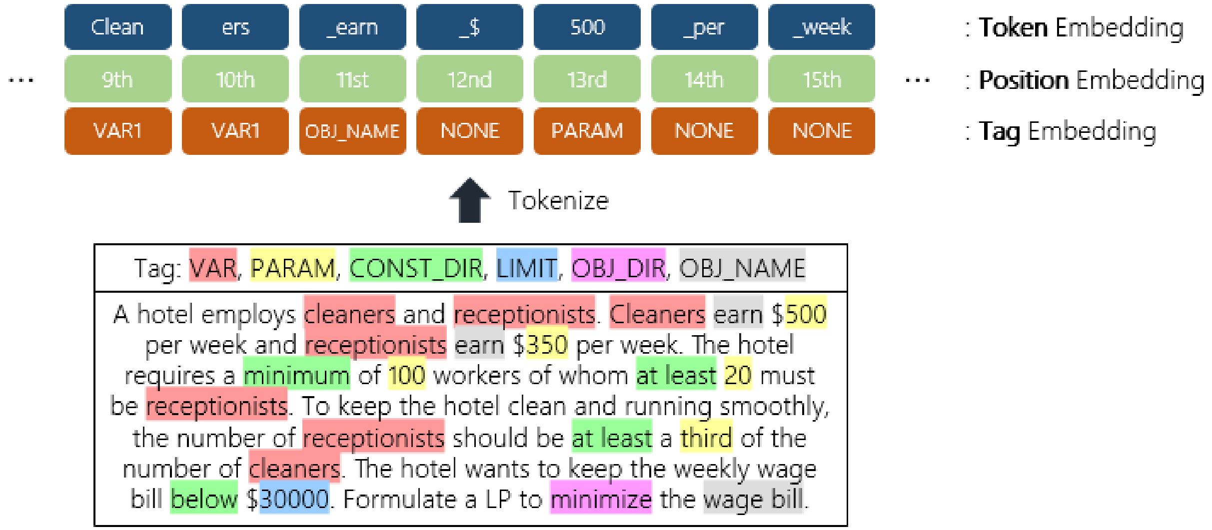
The number of train trips must not exceed the number of truck trips.	→ Augment	The number of train trips must exceed the number of truck trips.
x ≤ y ;		x ≥ y ;

POSTECH

Entity Tag Embedding and Embedding Scaling

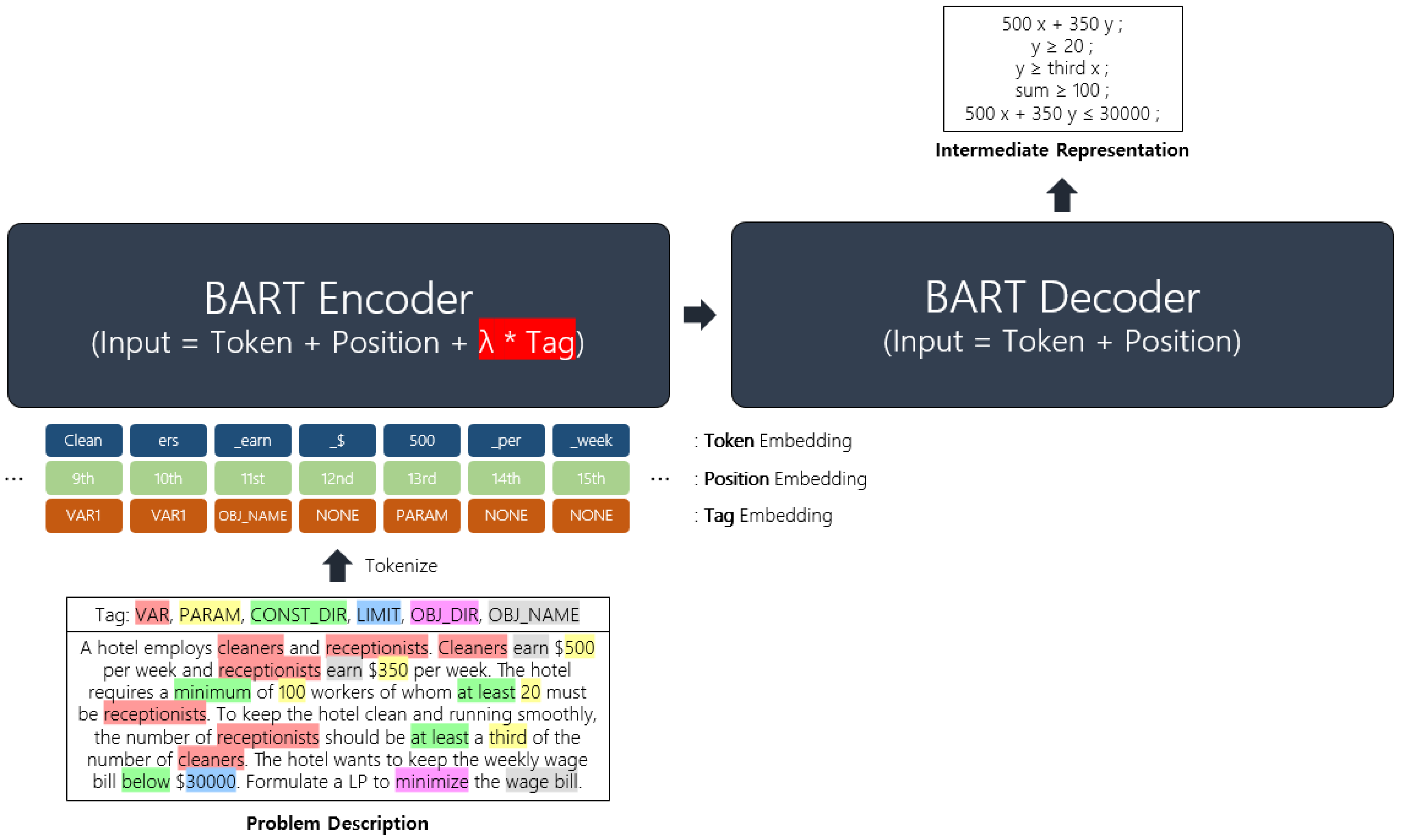
- We finetune pretrained $\text{BART}_{\text{large}}$ for auto-formulation task.
  - (1) token embeddings: $E_{w_1}^{tok}, E_{w_2}^{tok}, \dots, E_{w_L}^{tok} \in \mathbb{R}^d$
  - (2) position embeddings: $E_1^{pos}, E_2^{pos}, \dots, E_L^{pos} \in \mathbb{R}^d$
  - (3) entity tag embeddings: $E_{t_1}^{tag}, E_{t_2}^{tag}, \dots, E_{t_L}^{tag} \in \mathbb{R}^d$ → initialize to 0s

- We use an embedding scaling hyperparameter λ for entity tag embedding.
  - (1) l-th input embedding: $E_{w_l}^{tok} + E_l^{pos} + \lambda E_{t_l}^{tag}$.



Problem Description

Method Overview



POSTECH

Experiments

- We use declaration-level mapping accuracy:

$$Accuracy = 1 - \frac{\sum_{i=1}^{N} FP_i + FN_i}{\sum_{i=1}^{N} D_i}$$

- Hyperparameters:

Hyperparameters	
Batch Size	16
Optimizer	AdamW
Learning Rate	5e-5
Weight Decay	1e-5
LR Scheduler	Cosine Annealing
Max Norm (Gradient Clipping)	1.0
Num Epochs	100
Generation	Beam Search
Num Beams	4

POSTECH

Ablation Study

- The results demonstrate the effectiveness of the proposed techniques.
  (1) λ: embedding scaling weight for entity tag
  (2) p: probability of reversing constraint direction (for data augmentation)

Hyperparameter			Validation Accuracy	
BART Size	λ	p		
Base	0	0	0.5513	
Large	0	0	0.7718	→ Larger pretrained model
Large	1	0	0.8000	→ Adding entity tag embedding
Large	5	0	0.8692	→ Adjusting weight of entity tag embedding
Large	5	0.3	0.8846	→ Data augmentation

Result

- We placed 2nd in NL4Opt competition subtask 2.

Team	Test Accuracy
UIUC-NLP	0.899
Sjang	0.878
Long	0.867
PingAn-zhiniao	0.866
Infrrd AI Lab	0.780
KKKKKi	0.634

POSTECH

Thank You