**Algorithm: Proxy-Tuning of Language Model**

/\* Applies proxy-tuning to adjust logits of a target model in an NLP task \*/

**Input**: input\_ids, large\_model\_base M, small\_model\_tuned M+, small\_model\_untuned M-,

**Output**: generated\_text, a sequence of tokens

**Hyperparameters**: max\_length (maximum generation length), n (number of tokens to generate)

**Parameters**: Θ includes all parameters for the large\_model\_base M, small\_model\_tuned M+, and small\_model\_untuned M-.

1. Initialize generated\_tokens as an empty list

2. Encode input\_text into input\_ids using tokenizer Θ\_tokenizer

// Perform token-wise proxy-tuning and text generation

3. For t in [1, ..., n]:

a. Obtain large\_model\_base, small\_model\_tuned, and small\_model­\_untuned logits:

i. large\_base\_logits ← large\_model\_base M (input\_ids). logits with parameters Θ\_large\_model\_base

ii. small\_tuned\_logits ← small\_model\_tuned M+ (input\_ids). logits with parameters Θ\_small\_model\_tuned

iii. small\_untuned\_logits ← small\_model\_untuned M- (input\_ids). logits with parameters Θ\_small\_model\_untuned

b. Proxy-tuning adjustment:

i. Δlogit\_offsets ← small\_tuned\_logits - small\_untuned\_logits

ii. logits' ← large\_base\_logits + Δlogit\_offsets

c. Normalize the logits for next token prediction:

i. predictions ← softmax (logits', axis=-1)

d. Select the next token:

i. next\_token\_id ← argmax(predictions)

ii. Append next\_token\_id to generated\_tokens

e. Update input\_ids with next\_token\_id for the next iteration

4. Decode the sequence of generated\_tokens into text using Θ\_tokenizer

5. **Return** generated\_text