

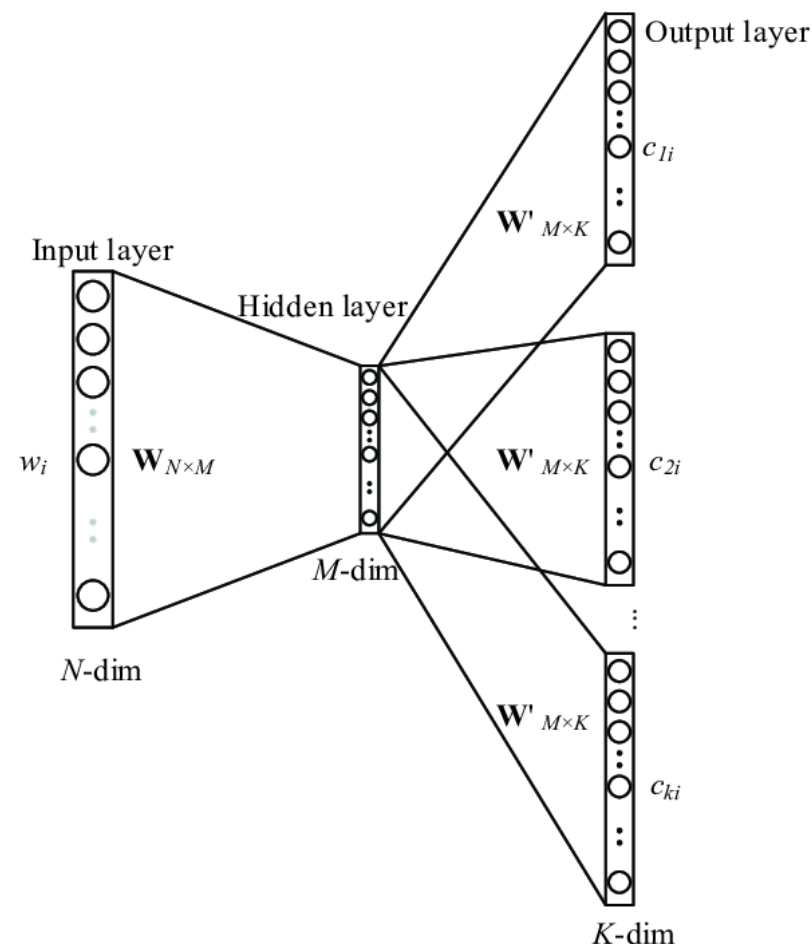
Language Models

Materials MS DL Course of Boris Zubarev @bobazooba

Word Embeddings

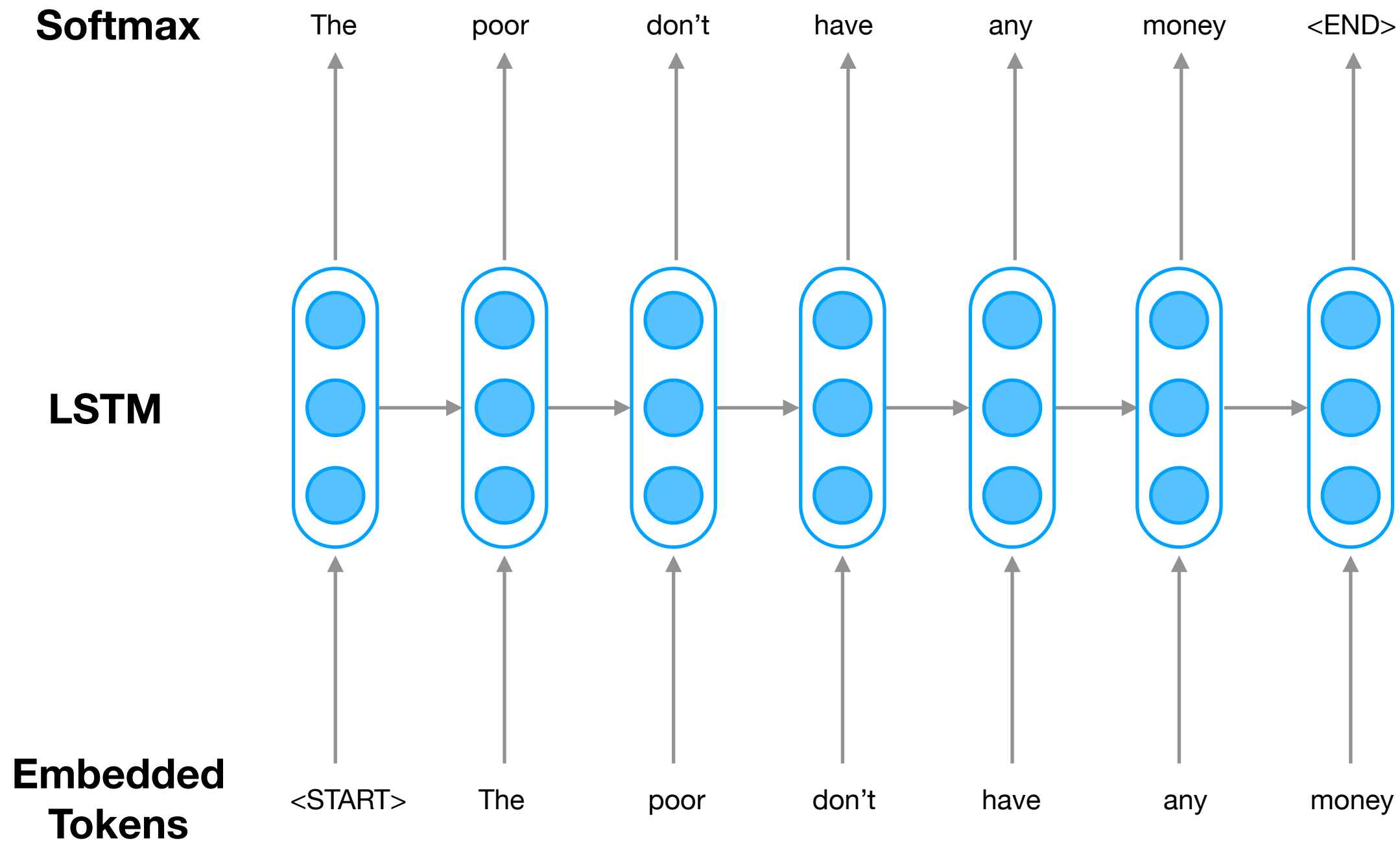
w2v, glove, etc

- Just key—value storage at inference
- Don't change from relationships with other words in the current text



Language Model

Training



Language Model

Prediction

Softmax

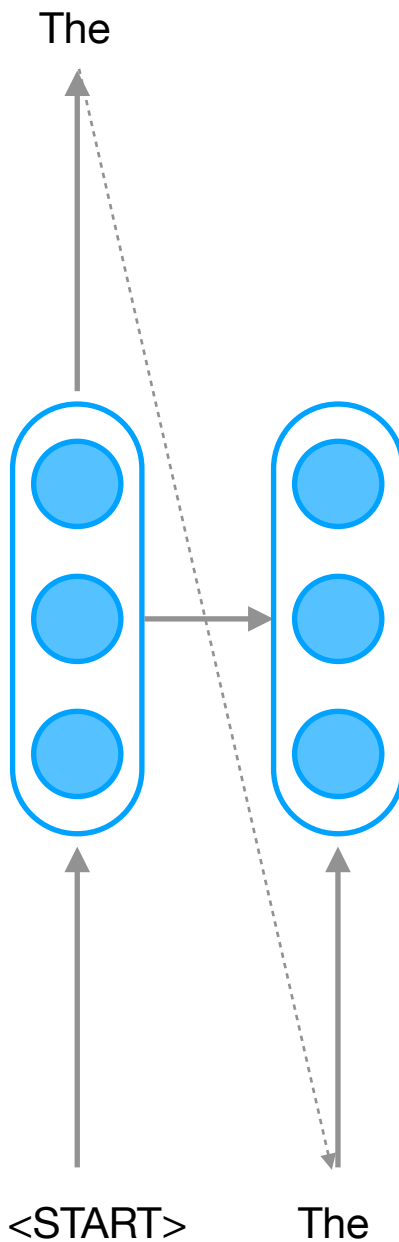
The

LSTM

**Embedded
Tokens**

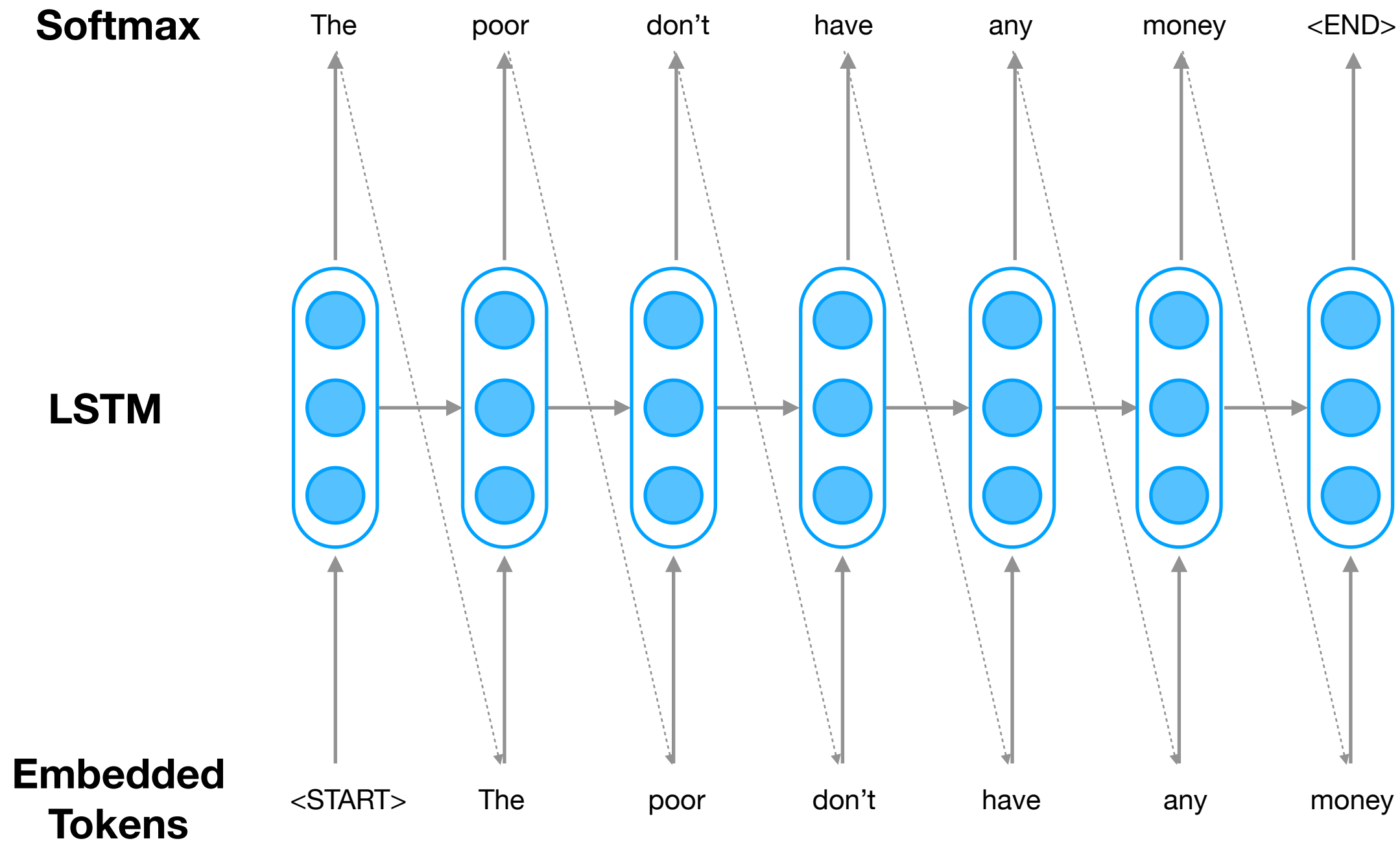
<START>

The

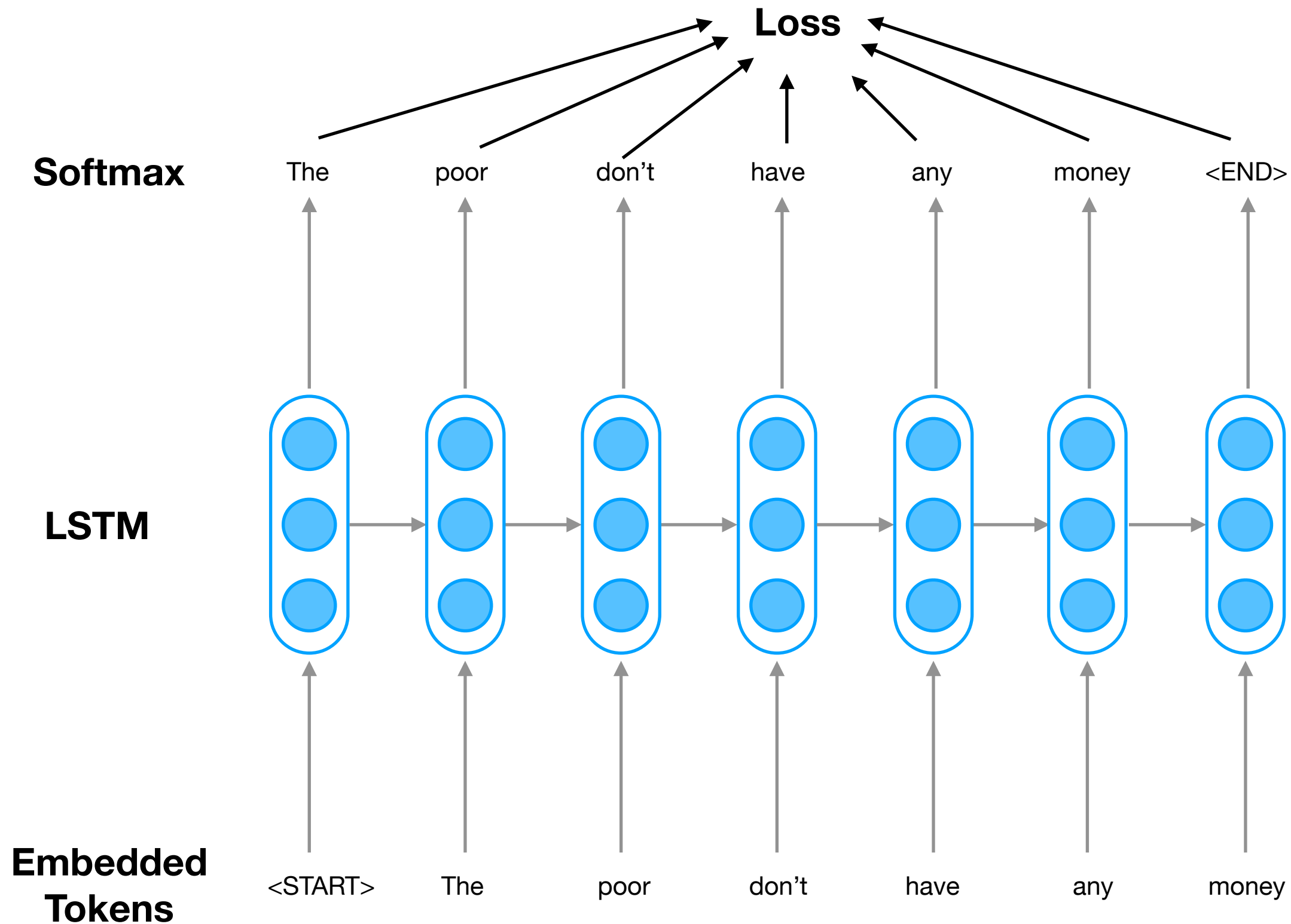


Language Model

$$p(\mathbf{x}) = \prod_i p(x|x_{<i}) = p(x_0)p(x_1|x_0)p(x_2|x_0, x_1)\dots$$



Language Model



Language Model

X

<START>

The

poor

Y

don't

Language Model

X

<START>

The

poor

don't

Y

have

Language Model

X

<START>

The

poor

don't

have

Y

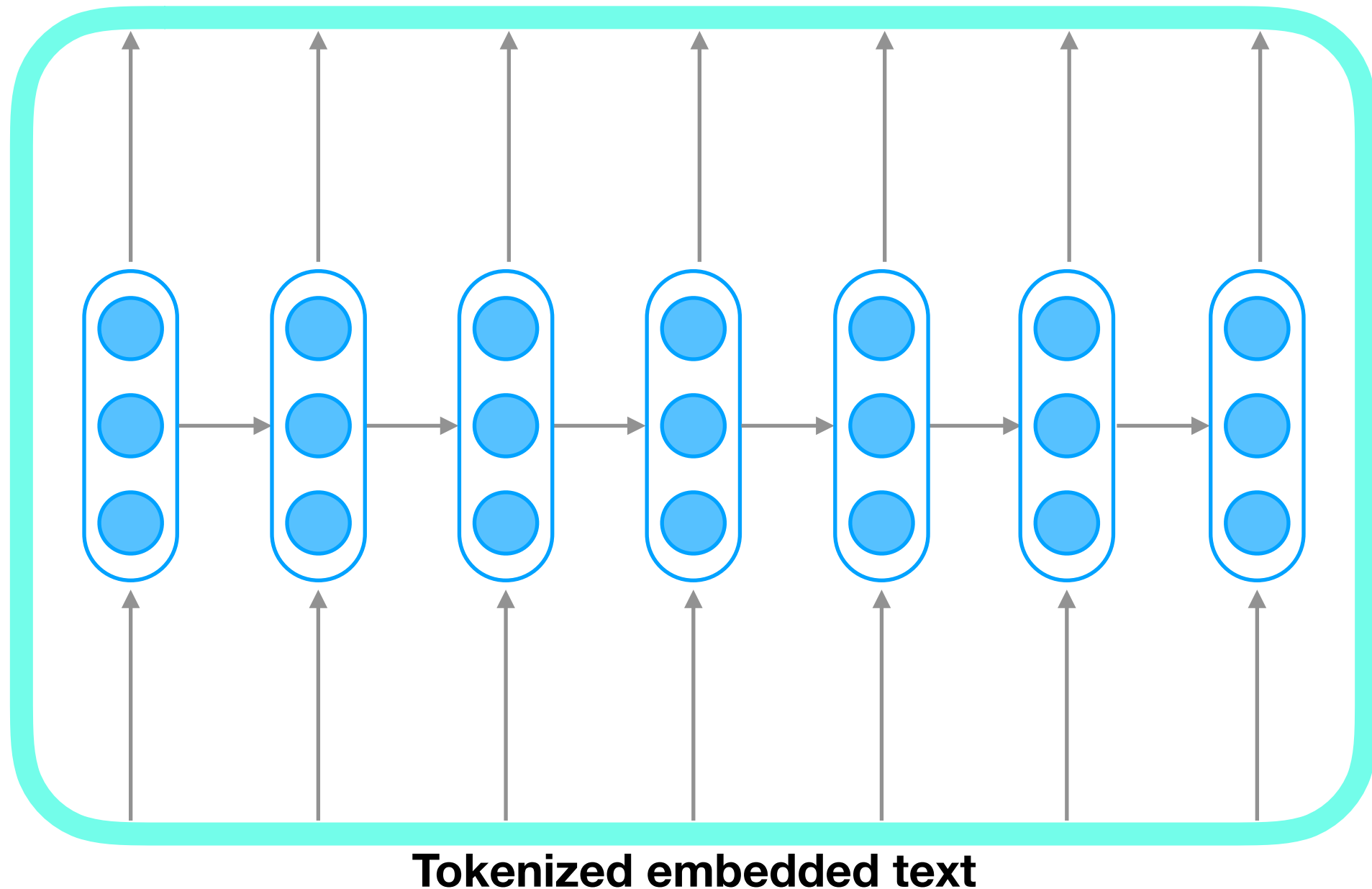
any

NLP LM Transfer Learning

Your typical classifier

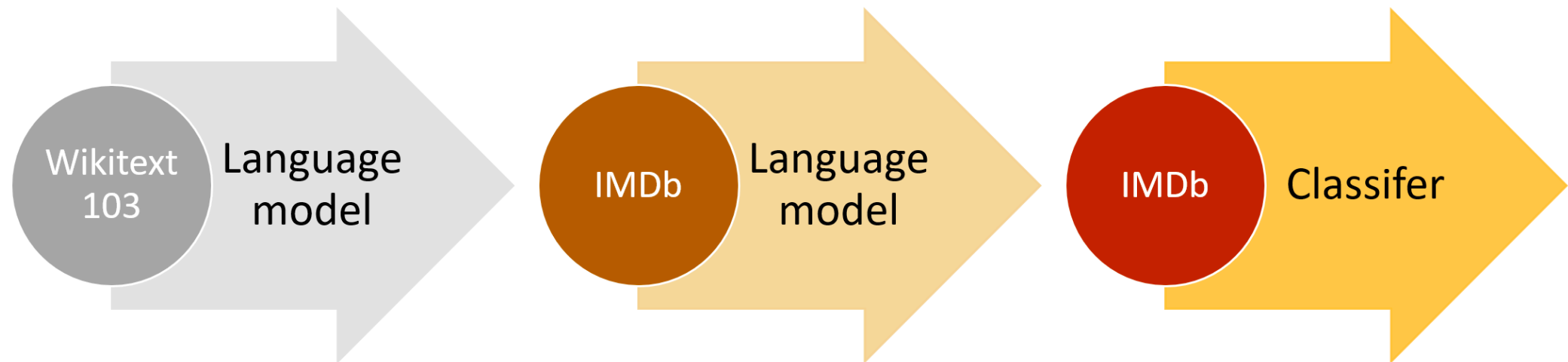
Contextualized word embeddings

Frozen
LSTM
Language
Model

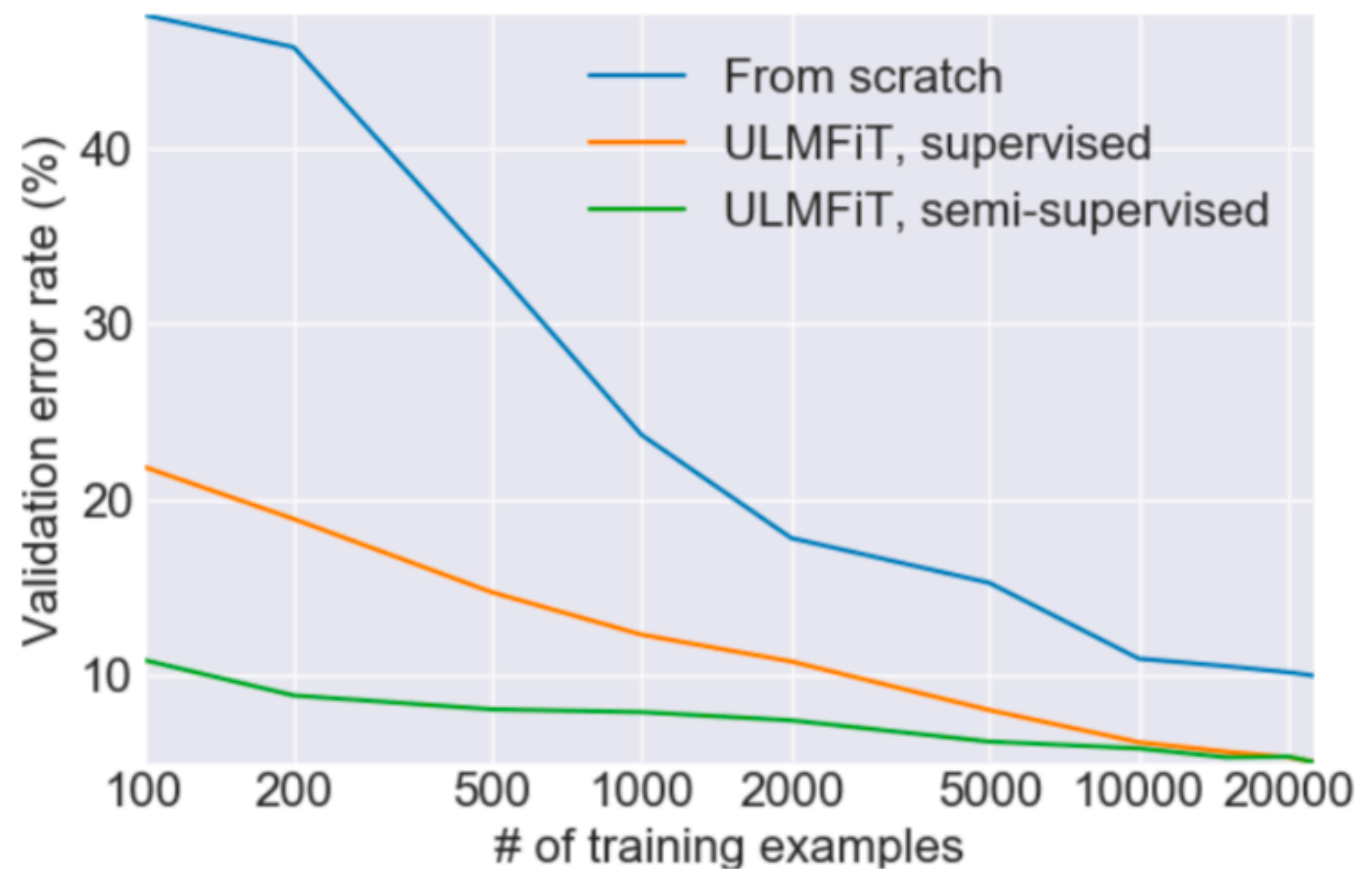
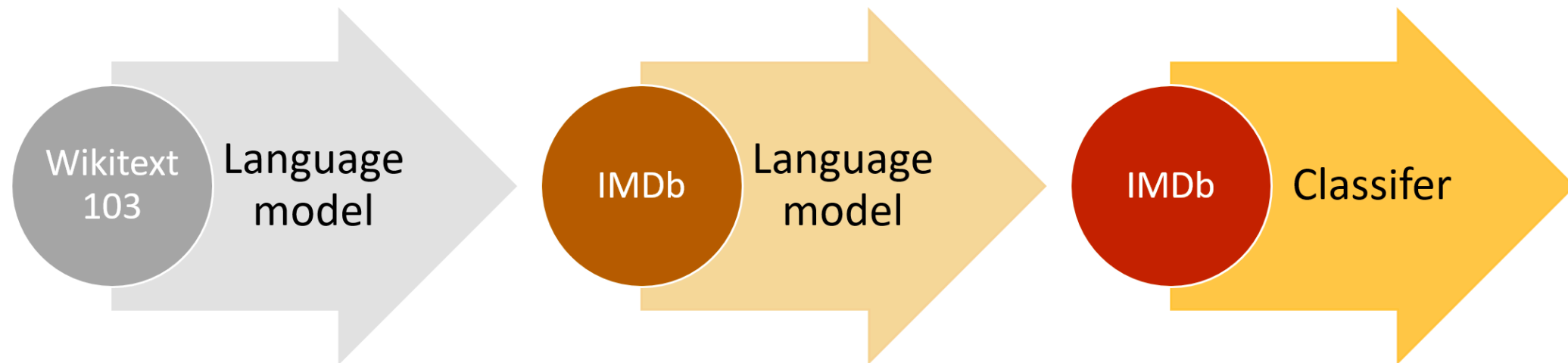


ULMFiT

ULMFiT



ULMFiT



ULMFiT

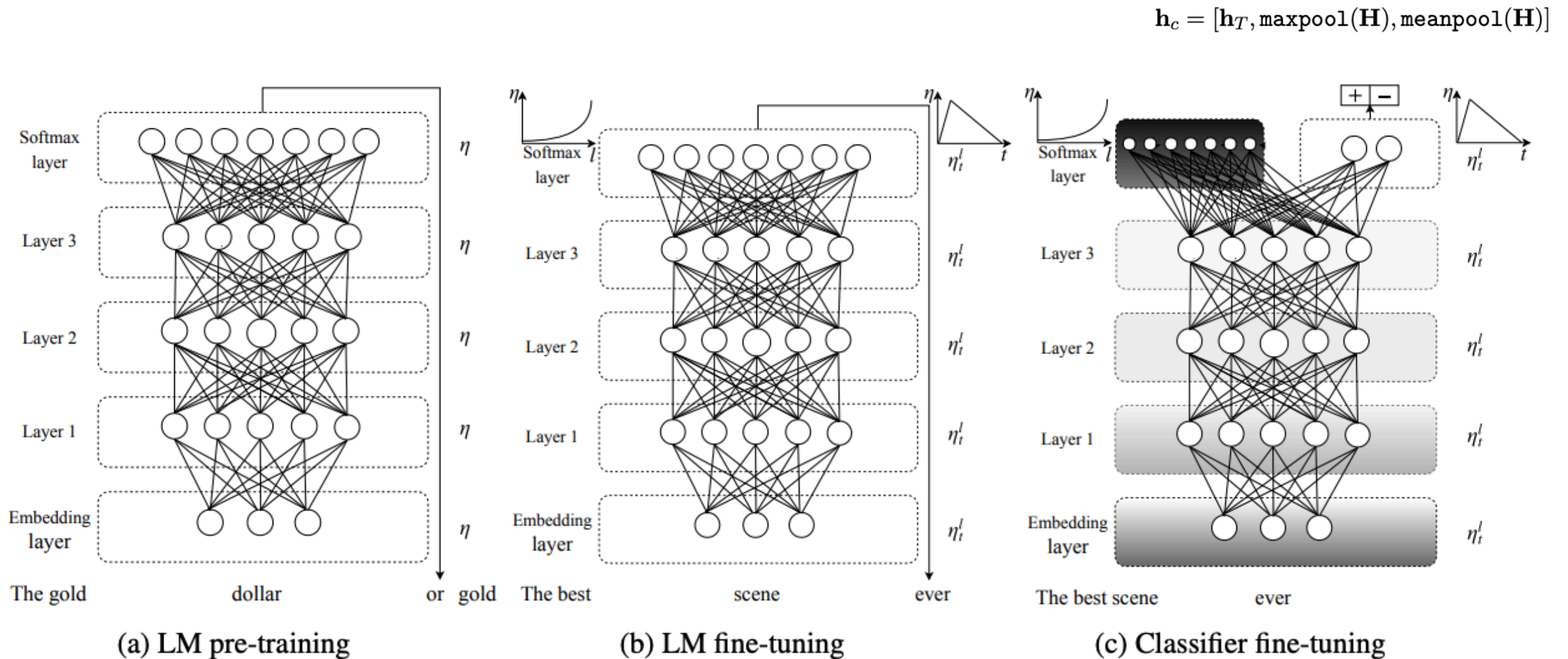


Figure 1: ULMFiT consists of three stages: a) The LM is trained on a general-domain corpus to capture general features of the language in different layers. b) The full LM is fine-tuned on target task data using discriminative fine-tuning (*Discr*) and slanted triangular learning rates (STLR) to learn task-specific features. c) The classifier is fine-tuned on the target task using gradual unfreezing, *Discr*, and STLR to preserve low-level representations and adapt high-level ones (shaded: unfreezing stages; black: frozen).

Scheduling

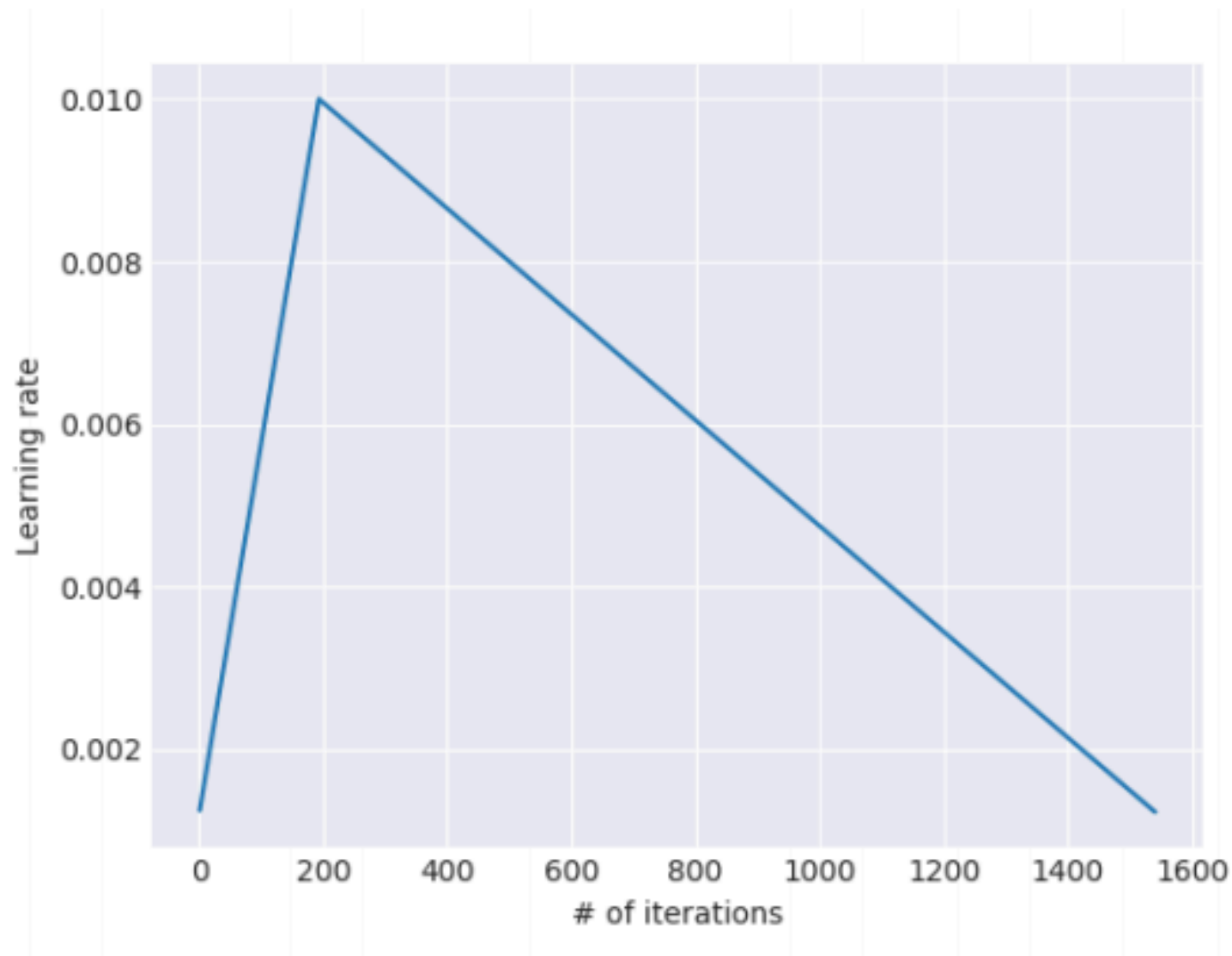
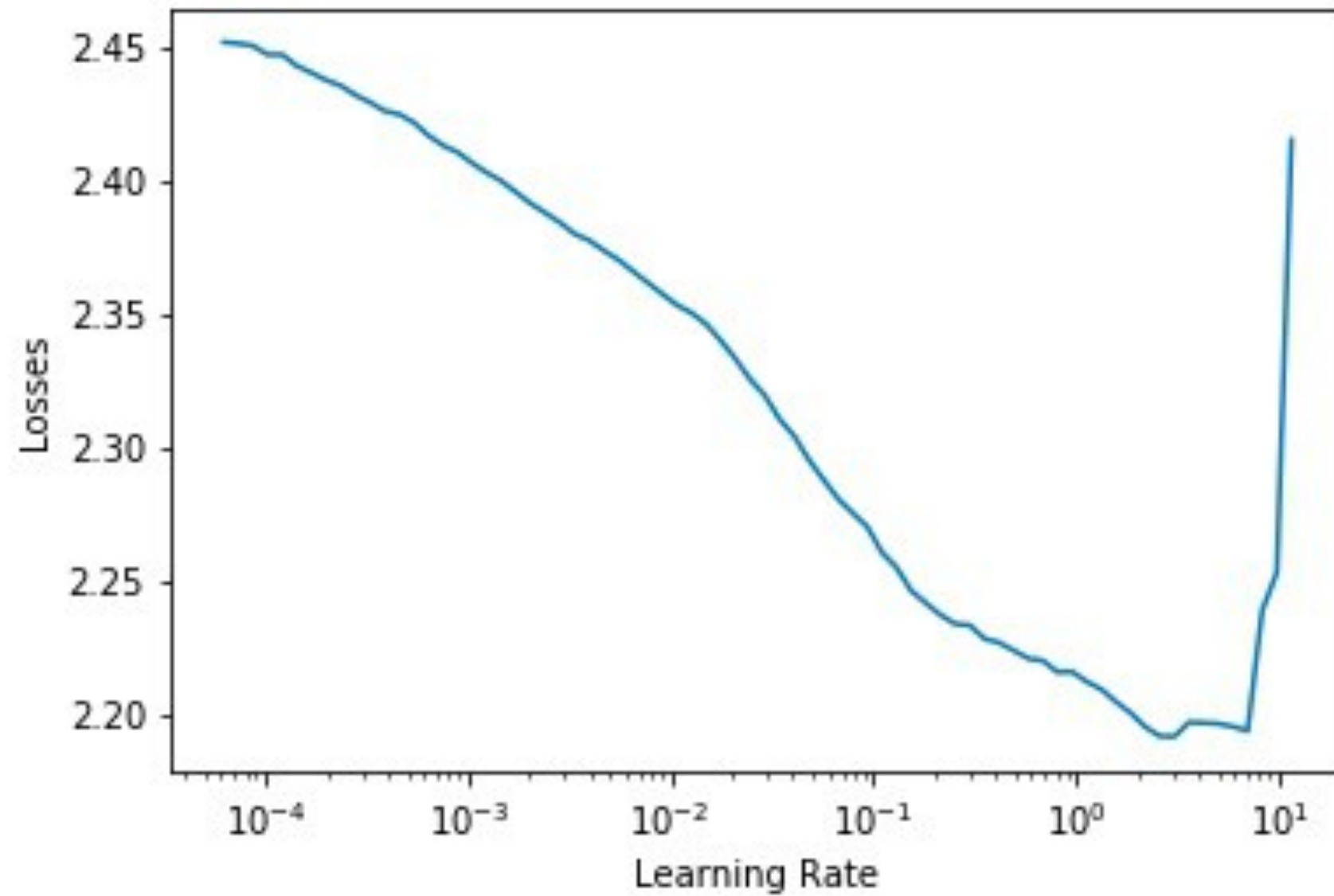


Figure 2: The slanted triangular learning rate schedule used for ULMFiT as a function of the number of training iterations.

LR Finder



ULMFiT

Gradual Unfreezing (Catastrophic Forgetting)

$$\mathbf{h}_c = [\mathbf{h}_T, \text{maxpool}(\mathbf{H}), \text{meanpool}(\mathbf{H})]$$

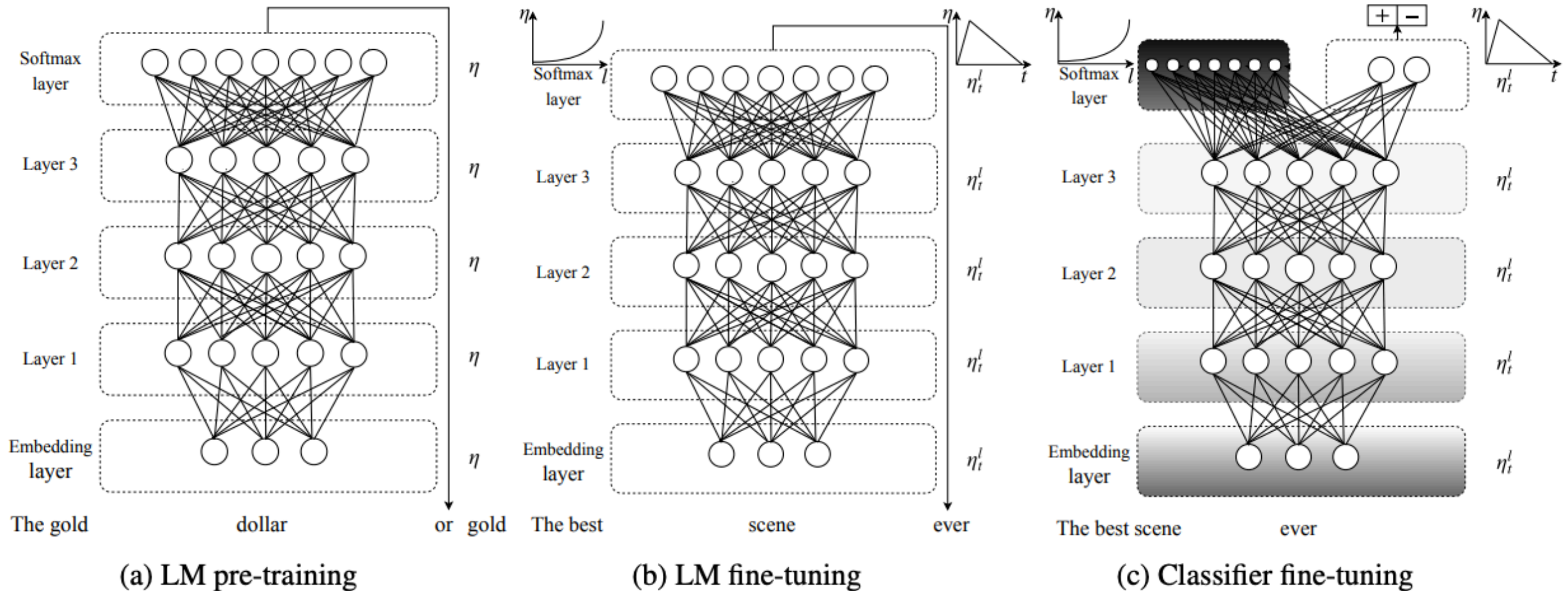


Figure 1: ULMFiT consists of three stages: a) The LM is trained on a general-domain corpus to capture general features of the language in different layers. b) The full LM is fine-tuned on target task data using discriminative fine-tuning (*Discr*) and slanted triangular learning rates (STLR) to learn task-specific features. c) The classifier is fine-tuned on the target task using gradual unfreezing, *Discr*, and STLR to preserve low-level representations and adapt high-level ones (shaded: unfreezing stages; black: frozen).

ULMFiT

Classifier

$$\mathbf{h}_c = [\mathbf{h}_T, \text{maxpool}(\mathbf{H}), \text{meanpool}(\mathbf{H})]$$

```
(1): PoolingLinearClassifier(  
  (layers): Sequential(  
    (0): BatchNorm1d(900, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)  
    (1): Dropout(p=0.4)  
    (2): Linear(in_features=900, out_features=50, bias=True)  
    (3): ReLU(inplace)  
    (4): BatchNorm1d(50, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)  
    (5): Dropout(p=0.1)  
    (6): Linear(in_features=50, out_features=2, bias=True)  
  )  
)
```

AWD-LSTM

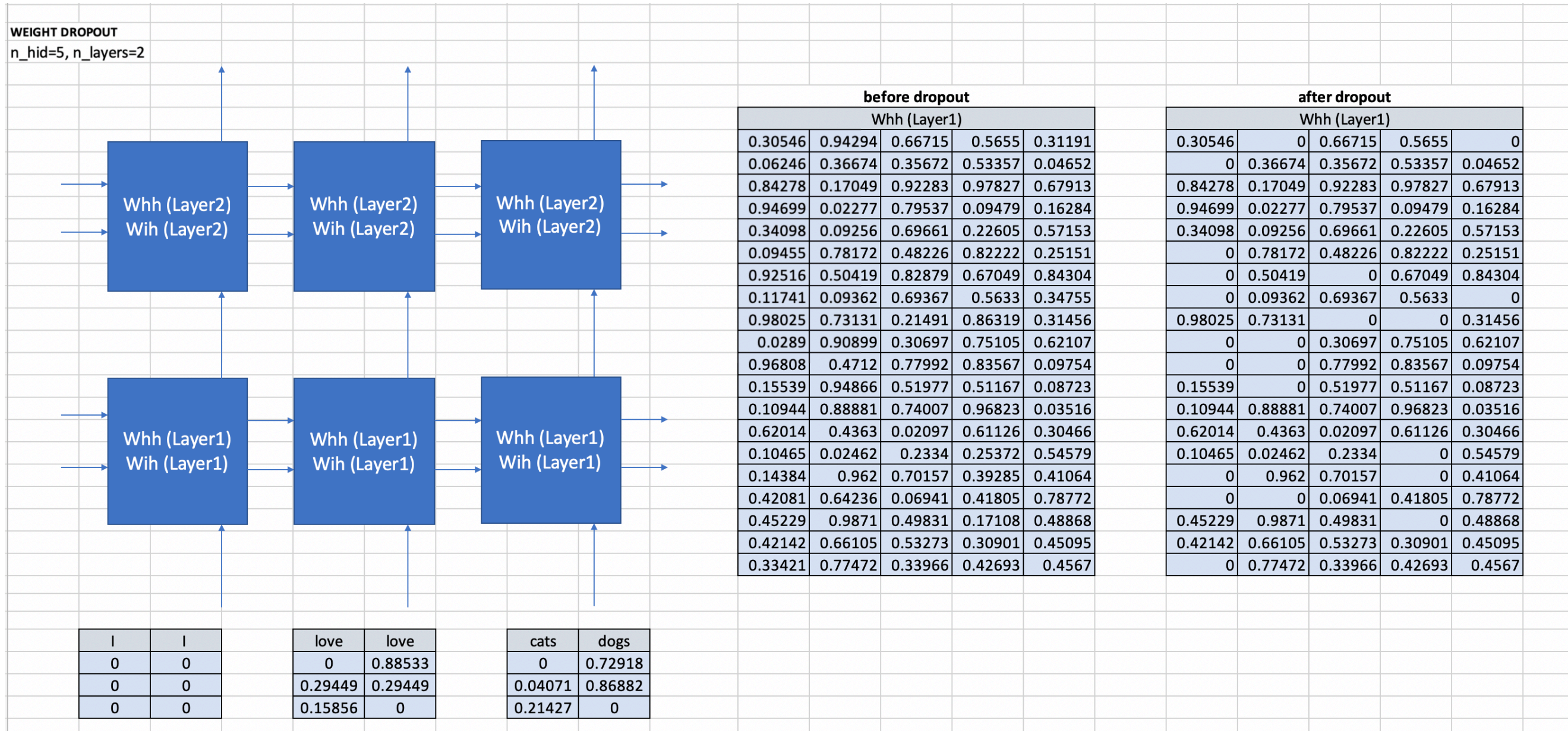
ENCODER DROPOUT									
before dropout					after dropout				
token	d1	d2	...		token	d1	d2	...	
I	0.399	0.75379	0.62616		I	0	0	0	
love	0.88533	0.29449	0.15856		love	0.88533	0.29449	0.15856	
cats	0.48927	0.04071	0.21427		cats	0.48927	0.04071	0.21427	
dogs	0.72918	0.86882	0.77136		dogs	0.72918	0.86882	0.77136	

AWD-LSTM

ENCODER DROPOUT									
before dropout					after dropout				
token	d1	d2	...		token	d1	d2	...	
I	0.399	0.75379	0.62616		I	0	0	0	
love	0.88533	0.29449	0.15856		love	0.88533	0.29449	0.15856	
cats	0.48927	0.04071	0.21427		cats	0.48927	0.04071	0.21427	
dogs	0.72918	0.86882	0.77136		dogs	0.72918	0.86882	0.77136	

INPUT DROPOUT										
batch: [I love cats, I love dogs]										
	before dropout					after dropout				
	I	0	0	0		I	0	0	0	
	love	0.88533	0.29449	0.15856		love	0	0.29449	0.15856	
	cats	0.48927	0.04071	0.21427		cats	0	0.04071	0.21427	
	before dropout					after dropout				
	I	0	0	0		I	0	0	0	
	love	0.88533	0.29449	0.15856		love	0.88533	0.29449	0	
	dogs	0.72918	0.86882	0.77136		dogs	0.72918	0.86882	0	

AWD-LSTM



AWD-LSTM

