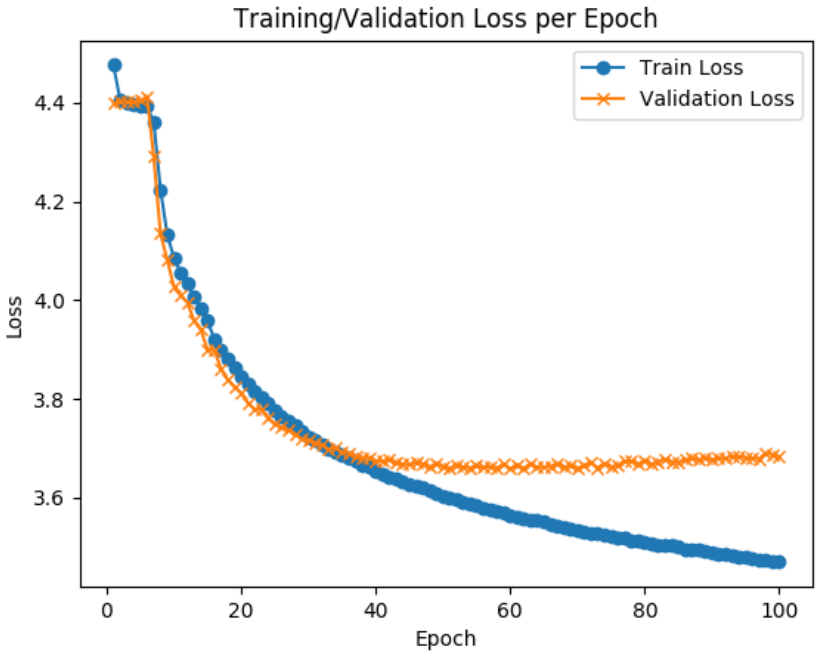
Mile Stone 2
Implement baseline

Question to answer

- Read Me
  - Dataset link
  - How to import data
  - Additional library (OpenCV?, Keras?)
  - Other environment setting
- Training process
  - Goal
  - Network Structure
- Result
  - Table
    - for loss or acc@k or other?
    - After cross-validation
  - Graph for loss/epoch or acc/epoch
  - ScreenShot for program

Model type	dim	Flat hit@k (%)				Hierarchical precision@k			
		1	2	5	10	2	5	10	20
Softmax baseline	N/A	**55.6**	**67.4**	**78.5**	**85.0**	0.452	0.342	0.313	0.319
DeViSE	500	53.2	65.2	76.7	83.3	0.447	**0.352**	**0.331**	**0.341**
	1000	54.9	66.9	78.4	**85.0**	**0.454**	0.351	0.325	0.331
Random embeddings	500	52.4	63.9	74.8	80.6	0.428	0.315	0.271	0.248
	1000	50.5	62.2	74.2	81.5	0.418	0.318	0.290	0.292
Chance	N/A	0.1	0.2	0.5	1.0	0.007	0.013	0.022	0.042



Rule

- One question for one section!
- Deadline: 11/10 23:59
- Format: ICML 2017 (using latex)
- Naming: Team01_MS2.zip
  - Include Team01_MS2.pdf and code (naming is unnecessary) , dataset is unnecessary
- 0 point for delay

Grading for Lecture Note

- Graph&equation: 80%
  - -5% if lost one
  - -2.5% if screenshot equation each
- Additional Supplement: 20%
  - Derivation will get 5% each
  - Supplement knowledge will get 5% each

Grading for HW1

- Upload on ILMS: 40%
- Accuracy: 30%
  - below 34% will get all
  - Above 1% will -1%
- Explain the graph: 30%
  - Textbook solution: explain the relation of matrix will get all.
  - Other solution: using gradient decent or other solution, must also explain your complicated graph.
- Bouns: 20%
  - Greater than baseline: 10%
  - Explain how and comment: 10%