

Επεξεργασία Φωνής και Φυσικής Γλώσσας

Προπαρασκευή 3ου Εργαστηρίου - Χειμερινό Εξάμηνο 2019-20

Βοηθητικό Υλικό

Επίσημοι Οδηγοί

Αρχικά πριν ξεκινήσετε να υλοποιείτε τα ερωτήματα του εργαστηρίου, θα πρέπει να μελετήσετε τους εξής επίσημους οδηγούς:

Ξεκινήστε από τους γενικούς οδηγούς:

- [Deep Learning with PyTorch: A 60 Minute Blitz](#)
- [Learning PyTorch with Examples](#)

Και μετά συνεχίστε σε αυτούς που εστιάζουν σε NLP προβλήματα:

- [Deep Learning for NLP with Pytorch](#)
 - [Introduction to PyTorch](#)
 - [Deep Learning with PyTorch](#)
 - [Word Embeddings: Encoding Lexical Semantics](#)

Οδηγοί από Τρίτους

Πολύ αξιόλογοι είναι και οι οδηγοί από το μάθημα [CS230 Deep Learning](#) του Stanford.

- [Introduction to PyTorch Code Examples](#)

An overview of training, models, loss functions and optimizers

- [Named Entity Recognition Tagging](#)

Defining a Recurrent Network and Loading Text Data

Το άρθρο “[PyTorch - Basic operations](#)” είναι ένας πολύ καλό σημείο αναφοράς με τις βασικές πράξεις που μπορείτε να κάνετε με ταυστές στο PyTorch.

Επίσης, μπορείτε να έχετε σαν σημείο αναφοράς, την ακόλουθη λίστα με υλοποιημένα μοντέλα σε Pytorch: [Lists of tutorials & examples](#)

Για τα ερωτήματα σχετικά με την προετοιμασία των δεδομένων, ίσως σας φανεί χρήσιμος ο ακόλουθος οδηγός: https://pytorch.org/tutorials/beginner/data_loading_tutorial.html

Συμβουλές Ανάπτυξης - Fail Fast!

Όταν σχεδιάζετε νευρωνικά δίκτυα, είναι σημαντικό να μπορείτε γρήγορα να δοκιμάσετε κατά πόσο μία νέα λειτουργία που υλοποιήσατε δουλεύει ή όχι. Αυτό είναι πρόβλημα γιατί πολλές φορές θα πρέπει να μεσαλαβήσουν πολλά βήματα μέχρι να φτάσετε στο επίμαχο σημείο. Για τον λόγο αυτό καλό θα ήταν αρχικά να υλοποιείτε κάθε λειτουργία, σε ένα ξεχωριστό script και να ελέγχετε τα αποτελέσματα.

Για παράδειγμα:

```
import torch
import torch.nn as nn

# 2dim tensor.. aka a matrix
x = torch.randn(16, 5)

# this is the same as:
batch_size = 16
feature_size = 5
x = torch.randn(batch_size, feature_size)

# now let's try out some NN layer
output_size = 3
fc = nn.Linear(feature_size, output_size)
print(fc(x).shape)
```

Επίσης μπορείτε να δημιουργήσετε και πιο σύνθετα παραδείγματα, όπως για παράδειγμα χρησιμοποιώντας ένα Embedding layer.

```
import torch
import torch.nn as nn

batch_size = 16
max_length = 8
n_embeddings = 1000
embedding_size = 50

# create batch of random sequences of token ids
x = (torch.rand(batch_size, max_length) * n_embeddings).long()
print(x.shape)

embed = nn.Embedding(num_embeddings=n_embeddings,
                    embedding_dim=embedding_size)

embeddings = embed(x)
print(embeddings.shape)
```