

Διπλωματική εργασία

Τεχνικές μηχανικής μάθησης για πρόβλεψη εντολών σε περιβάλλοντα προγραμματισμού

Σπουδαστής:

Γερογιάννης Μιχαήλ

Επιβλέπων καθηγητής:

Χατζηγεωργίου Αλέξανδρος

28-02-2020



**University of
Macedonia**
MSc in
Applied
Informatics

Στόχος διπλωματικής εργασίας

- Ανάδειξη πλεονεκτημάτων των νευρωνικών δικτύων στα πλαίσια του κλάδου της επεξεργασίας φυσικής γλώσσας.
- Δημιουργία εργαλείου αυτόματης συμπλήρωσης κώδικα.

Παρουσίαση του προβλήματος



```
1 import numpy as np
2 import m
```

matplotlib	module
math	module
multiprocessing	module
mock	module
mpl_toolkits	module
mako	module

Έννοιες που θα χρησιμοποιηθούν

- Νευρωνικά δίκτυα
- Αναδρομικά νευρωνικά δίκτυα
- Δίκτυα ανατροφοδοτούμενης μονάδας με πύλη

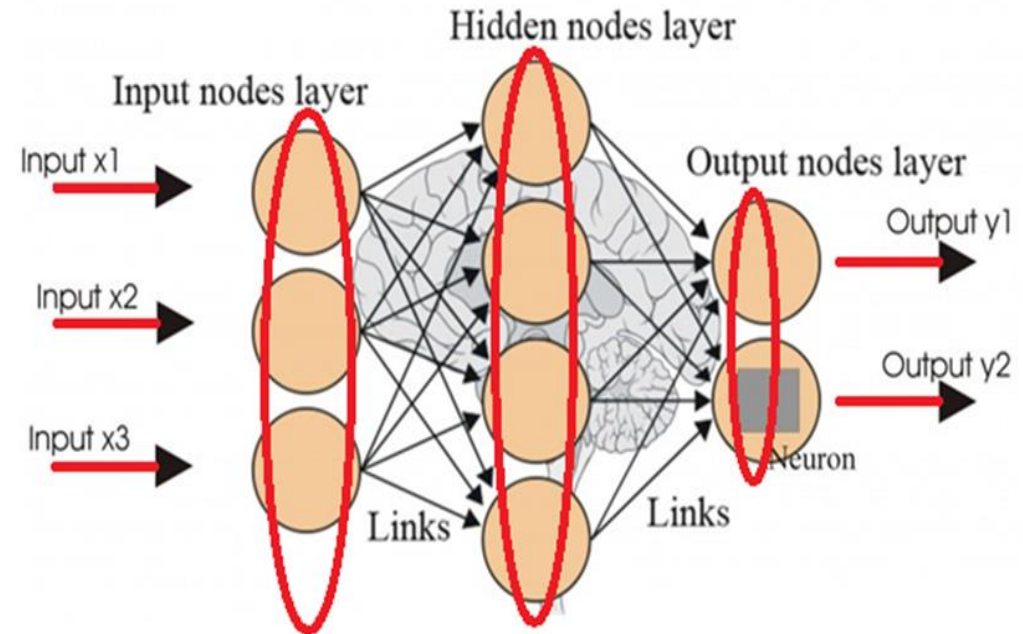
Βιβλιογραφική επισκόπηση

- Long short term memory networks (LSTM) – Li et al. [2017]
- Sparse pointer networks – Bhoopchand et al. [2016]
- Convolutional neural networks – Hinton et al.[2012]
- Abstract syntax tree – Liu et al. [2016]
- Δίκτυα ανατροφοδοτούμενης μονάδας με πύλη (GRU)

Νευρωνικό δίκτυο

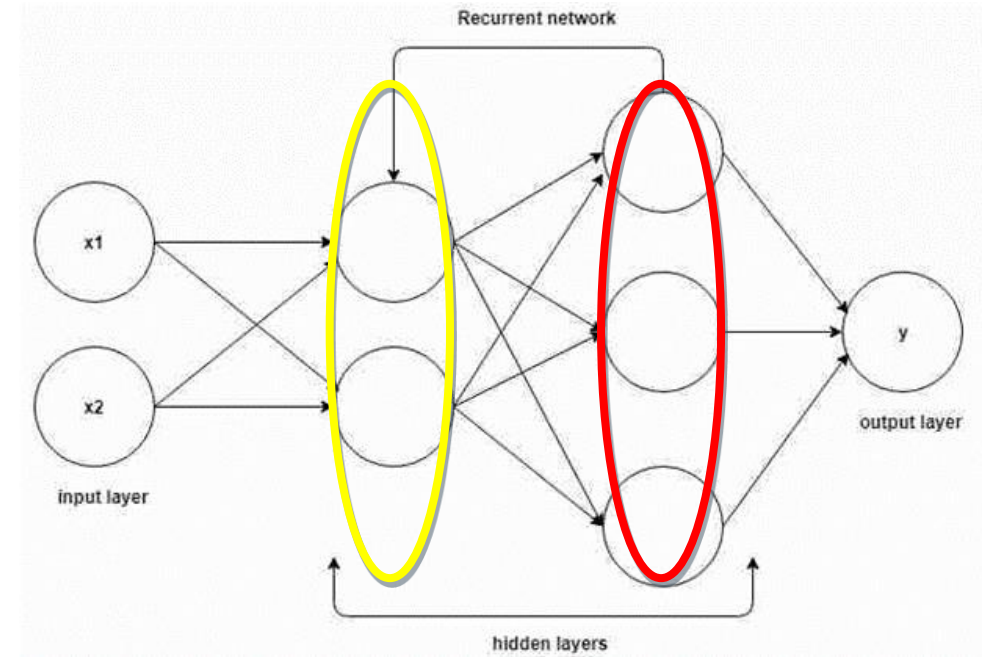
Χαρακτηριστικά νευρωνικού δικτύου

- Αποτελείται από νευρώνες.
- Οι νευρώνες συνδέονται μεταξύ τους.
- Κάθε σύνδεση έχει ένα συναπτικό βάρος.
- Δέχεται αριθμητικά δεδομένα εισόδου.
- Αδυναμία επεξήγησης του αποτελέσματος που παρέχουν.



Αναδρομικό νευρωνικό δίκτυο

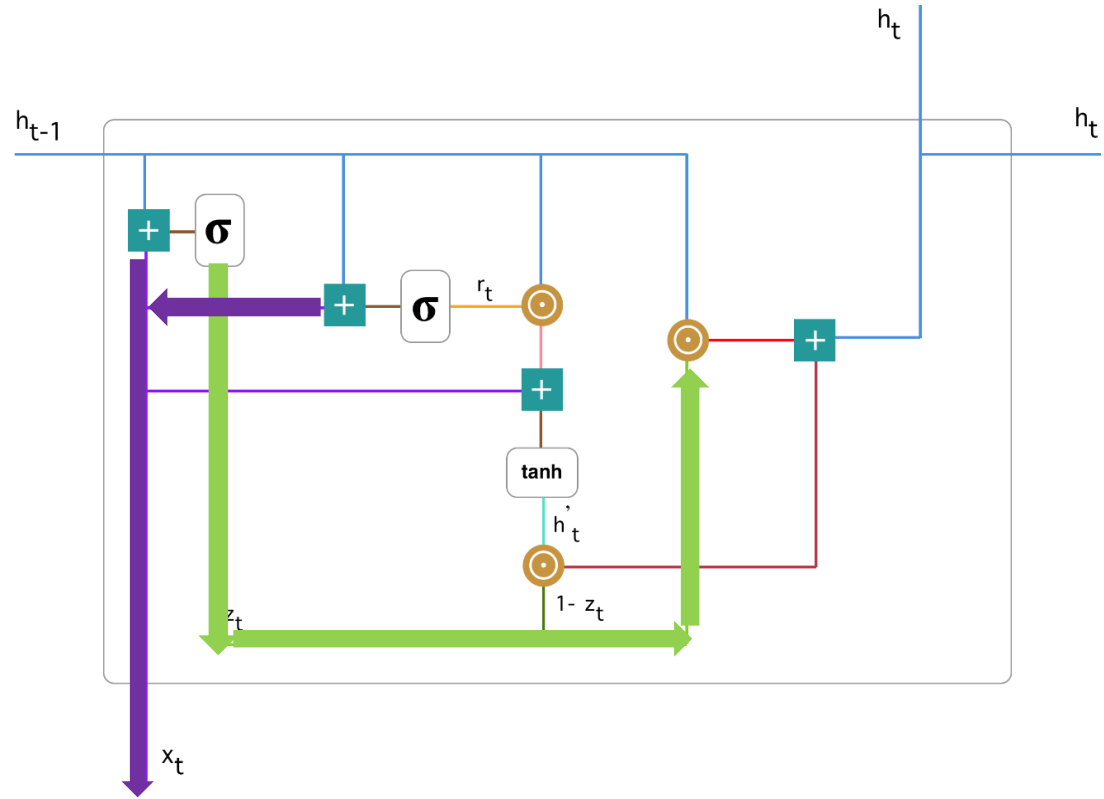
- Μια υποκατηγορία των νευρωνικών δικτύων.
- Χρησιμοποιούν βρόγχους ανατροφοδότησης.
- Δεν διαθέτουν αρκετή μνήμη.



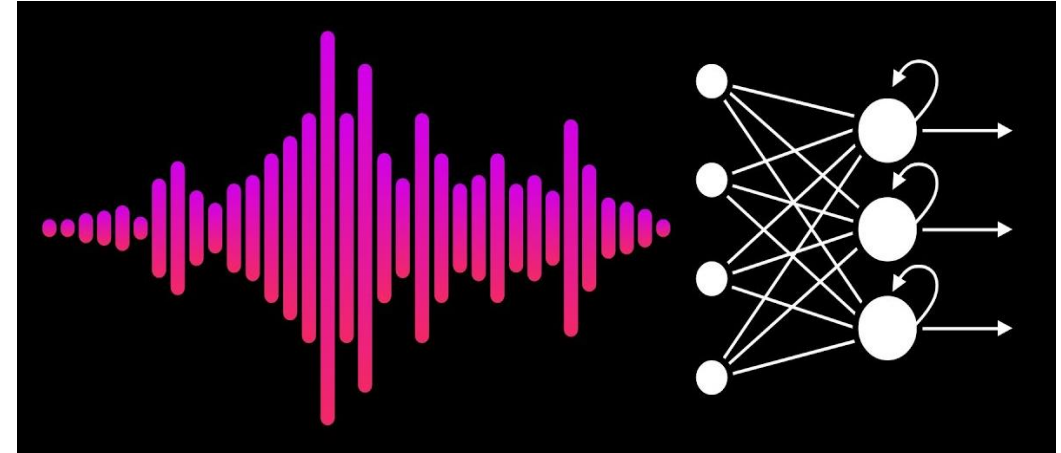
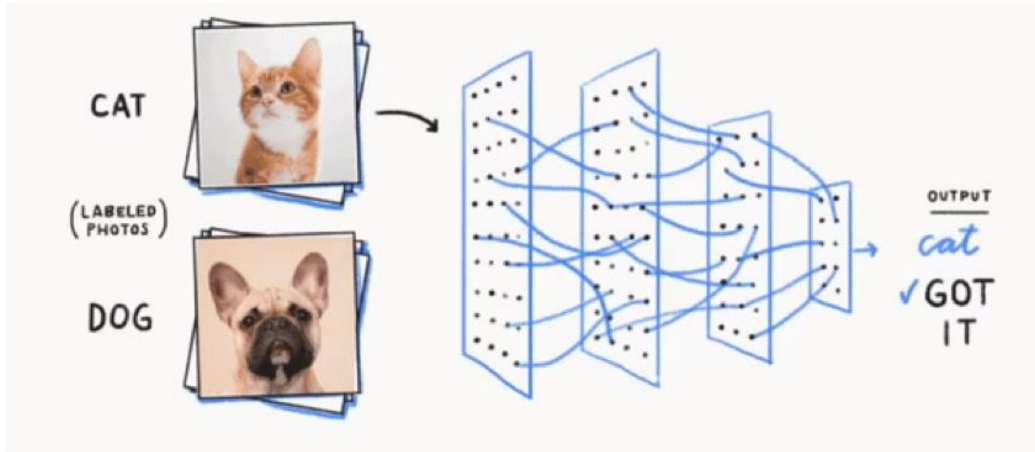
Δίκτυα ανατροφοδοτούμενης μονάδας με πύλη

Γιατί είναι σημαντικά:

- Προσφέρουν ένα είδος μνήμης (πύλες).
- Μέσω της πύλης ενημέρωσης αποφασίζουν ποιες πληροφορίες πρέπει να συνεχίσουν στα επόμενα στρώματα.
- Μέσω της πύλης επαναφοράς αποφασίζουν ποιες πληροφορίες δε πρέπει να ληφθούν υπόψιν.



Χρήσεις νευρωνικών δικτύων

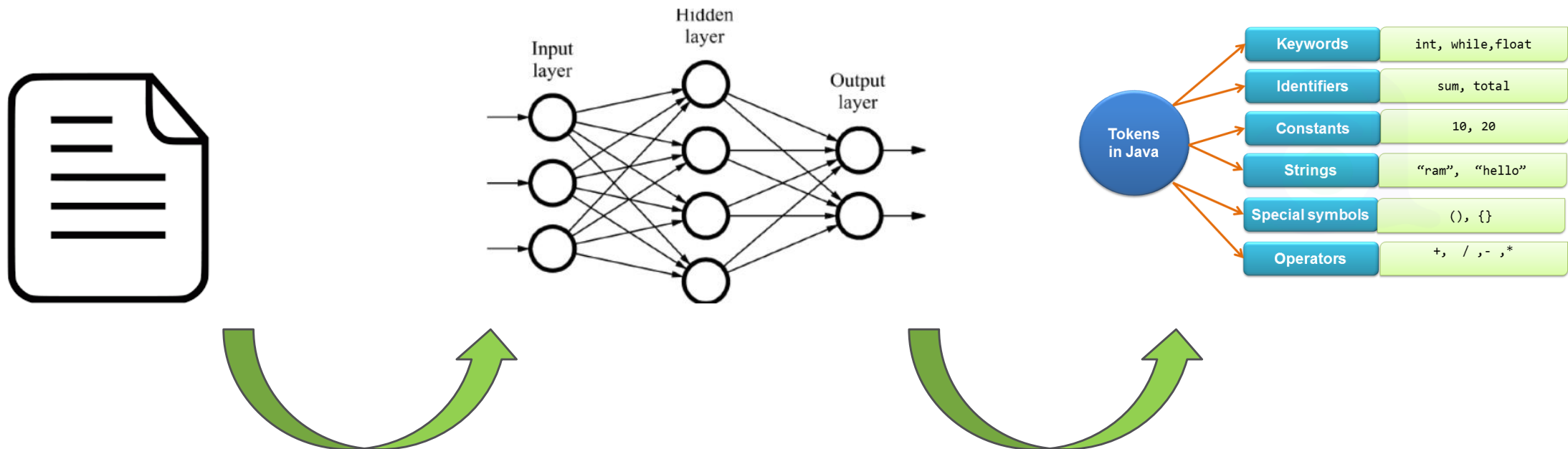


```
1 import numpy as np
2 import m
```

matplotlib	module
math	module
multiprocessing	module
mock	module
mpl_toolkits	module
mako	module

Παρουσίαση εφαρμογής

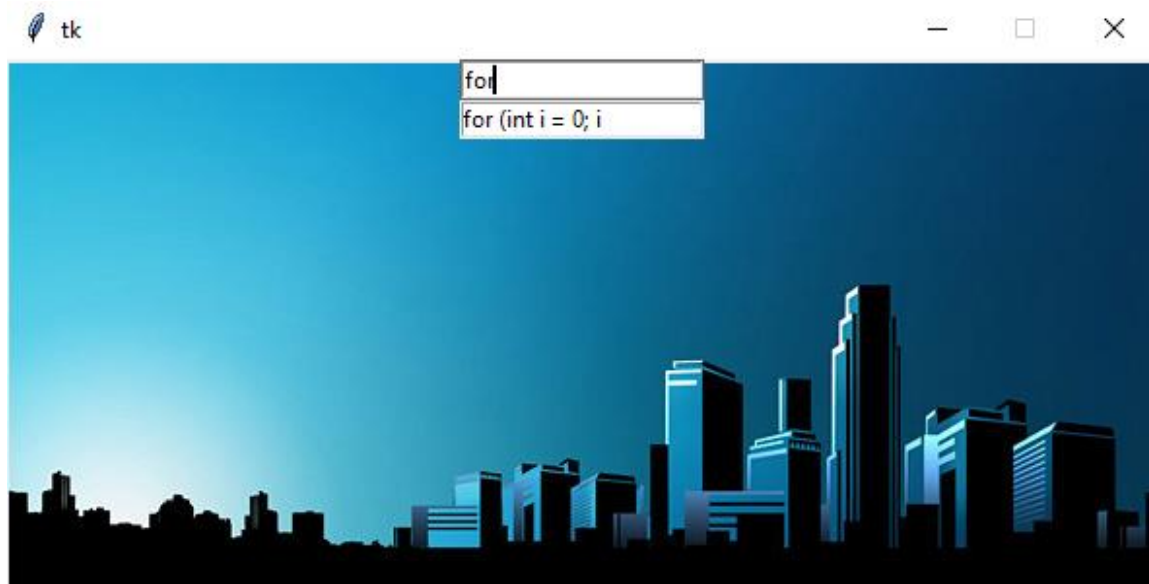
- Γλώσσα προγραμματισμού Python.
- Αρχείο κειμένου με projects γραμμένα σε Java για την εκπαίδευση του δικτύου (Input).
- Σύμβολα και λέξεις του συντακτικού της Java (Output).



Μέθοδος create_model

```
def create_model(predictors, label, max_sequence_len, total_words):  
  
    #GRU  
    model = Sequential()  
  
    model.add(Embedding(input_dim=total_words,output_dim= total_words, input_length=max_sequence_len - 1))  
  
    model.add(GRU(units=150, return_sequences=True ))  
    model.add(GRU(units=50))  
  
    model.add(Dense(units=total_words, activation="softmax"))  
    model.compile(loss="categorical_crossentropy", optimizer="adam", metrics=['acc'])  
    #earlystop = EarlyStopping(monitor='val_loss', min_delta=0, patience=5, verbose=1, mode='auto')  
    model.fit(predictors, label, batch_size=128, epochs=3, verbose=1)  
    print  
    model.summary()  
    return model
```

Πρόβλεψη με το token “*for*”



Ακρίβεια νευρωνικού δικτύου

Μετρική categorical_crossentropy:

- Αυξάνεται καθώς η πρόβλεψη αποκλίνει από τη πραγματική τιμή.

```
Epoch 1/3  
161444/161444 [=====] - 46s 288us/step - loss: 0.5468 - acc: 0.8272  
Epoch 2/3  
161444/161444 [=====] - 47s 294us/step - loss: 0.1806 - acc: 0.8978  
Epoch 3/3  
161444/161444 [=====] - 49s 303us/step - loss: 0.1792 - acc: 0.8981  
Model: "sequential_2"
```

Layer (type)	Output Shape	Param #
embedding_2 (Embedding)	(None, 9, 57)	3249
gru_3 (GRU)	(None, 9, 150)	93600
gru_4 (GRU)	(None, 50)	30150
dense_2 (Dense)	(None, 57)	2907

Total params: 129,906
Trainable params: 129,906
Non-trainable params: 0



Σύγκριση μεθόδων

- Προεπεξεργασία των δεδομένων

Έρευνα	Μέθοδος	Ακρίβεια μοντέλου
Παρούσα έρευνα	GRU	89%
Hellendoorn & Devanbu [2017]	n-gram	86.2%
Bhoopchand et al. [2016]	Sparse pointer networks	84.5%
Li et al. [2017]	LSTM	81%
Tran et al. [2016]	LSTM	69.2%

Συμπεράσματα

- Τα νευρωνικά δίκτυα είναι αρκετά κοντά στο να παρέχουν ακριβείς προτάσεις συμπλήρωσης κώδικα.
- Η επιλογή μοντέλου νευρωνικού δικτύου καθώς και τα δεδομένα εκπαίδευσης είναι ζωτικής σημασίας.

Μελλοντικές επεκτάσεις

- Εφαρμογή chatbot στο περιβάλλον προγραμματισμού.
- Αναγνώριση ομιλίας εντός του περιβάλλοντος προγραμματισμού.

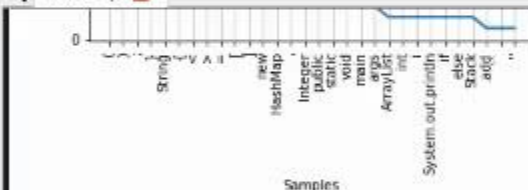
Usage

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our [tutorial](#)

Variable explorer	File explorer	Help
-------------------	---------------	------

Python console

 Console 1/4 

```
C:\Users\mgerogiannis\AppData\Local\Continuum\anaconda3\lib\site-packages\tensorflow_core\python\framework
\indexed_slices.py:433: UserWarning: Converting sparse IndexedSlices to a dense Tensor of unknown shape. This may
consume a large amount of memory.
```

"Converting sparse IndexedSlices to a dense Tensor of unknown shape."

```
Epoch 1/3
161444/161444 [-----] - 22s 135us/step - loss: 0.5322 - acc: 0.8322
Epoch 2/3
161444/161444 [-----] - 22s 135us/step - loss: 0.1798 - acc: 0.8977
Epoch 3/3
161444/161444 [-----] - 22s 139us/step - loss: 0.1765 - acc: 0.8988
Model: "sequential_2"
```

layer (type)	Output Shape	Param #
embedding_2 (Embedding)	(None, 9, 57)	3249
gru_3 (GRU)	(None, 9, 150)	93600
gru_4 (GRU)	(None, 50)	30150
dense_2 (Dense)	(None, 57)	2907

```
Total params: 129,906
Trainable params: 129,906
Non-trainable params: 0
```

Πέρασ παρουσίασης

Σας ευχαριστώ για τη προσοχή σας!