Practical work 11 – Recurrent Neural Networks with Keras (part 1)

Banfi Gregory (banfigre@students.zhaw.ch)
Benjamin Kühnis (bkuehnis@hsr.ch)

Exercice 1 **Human Activity Recognition**

We implemented the code of exercise one. SimpleRNN, L2 regularization, stacked RNN, confusion matrix, dropout and gradient clipping. We cannot fit the model after compiled it because some dimension error that can not be solved in time.

Exercice 2 **Language Classification by Last Names**

Model:

```
model = Sequential()
model.add(SimpleRNN(units=n_hidden, input_shape=(maxlen,len_alphabet)))
model.add(Dense(n_languages, activation='softmax'))

model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

Results for 100 epochs

hidden_layers 16 batch_size 200	`Test score: 0.7822955818959868` `Test accuracy: 0.7376283847289482`
hidden_layers 32 batch_size 200	`Test score: 0.7443639163543531` `Test accuracy: 0.7628384688043015`
hidden_layers 64 batch_size 200	`Test score: 0.9562373050009392` `Test accuracy: 0.7591036415957158`
hidden_layers 128 batch_size 200	`Test score: 1.0284423245991867` `Test accuracy: 0.7637721756203613`
hidden_layers 256 batch_size 200	`Test score: 0.8829030496399609` `Test accuracy: 0.760971055478275`
hidden_layers 128 batch_size 100	`Test score: 1.1420059480141747` `Test accuracy: 0.7507002801955247`
hidden_layers 128 batch_size 500	`Test score: 1.0721713529358836` `Test accuracy: 0.7366946780102817`

Model:

```
model = Sequential()
model.add(SimpleRNN(units=n_hidden, input_shape=(maxlen,len_alphabet)))
model.add(SimpleRNN(units=n_hidden, input_shape=(maxlen,len_alphabet)))
model.add(Dense(n_languages, activation='softmax'))

model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

Results for 100 epochs:

hidden_layers 64	`Test score: 1.0384861413392923`
batch_size 200	`Test accuracy: 0.7684407098119667`

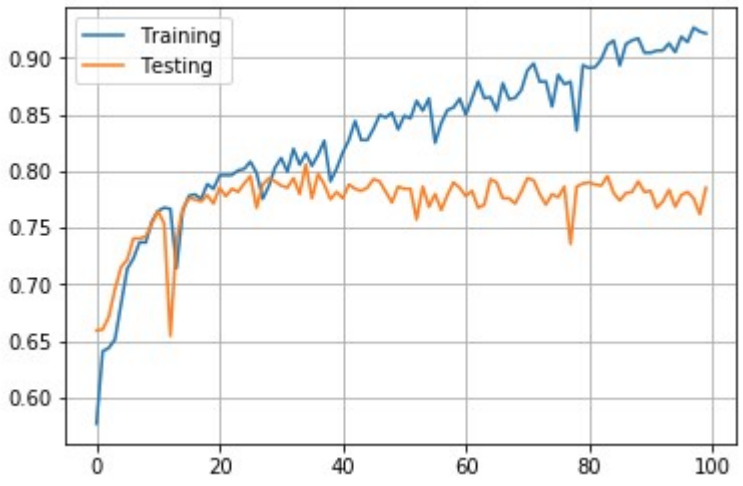
Model:

```
model = Sequential()
model.add(SimpleRNN(units=n_hidden, input_shape=(maxlen,len_alphabet)))
model.add(Dropout(0.2))
model.add(SimpleRNN(units=n_hidden, input_shape=(maxlen,len_alphabet)))
model.add(Dropout(0.2))
model.add(Dense(n_languages, activation='softmax'))

model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

hidden_layers 128	`Test score: 0.8648066375783734`
batch_size 200	`Test accuracy: 0.7983193277589191`
`hidden_layers 64`	`Test score: 0.8181301930121013`
`batch_size 200`	`Test accuracy: 0.7852474323340825`

We reached the same results seen in class thanks to the Dropout. The problem still is the few data provided that gives a large gap between training and test set predictions.



Exercice 3 **Sequence generation - startup names**

Generated company's names, with seed ''Adidas''

10 Epochs

```
e Software
"Bandablo Pibe
Stodetio
Bastcon Ound Matho In SEXE Co
Mohndens
SP Industrias
Fist Systems
PronaStaui
Mmerclind Fund
Pinnavoce
Shnectte
Raby'S Ltd
SodeFiraric Morchtin
Marlgityte PBR
SYS Technologies Lcadgiehtrnety Xpect
Share GmbH
I Devofmellatedia Corporation
ASH Styse
Morkited Medvoted Vi sticle bristifity Plications
Surple Biordgituct
Gratyk
Side Manker Technologies
Arevel
Strondedrovery Powere Energy, LLC"
ImpleMon Connerty
4RS
Pach Spart Health
Lingtambatd.com
EhivertUV Gompana S
```

50 Epochs

```
e
Stab Technologies
Nexwick
Victriz
TRUM, Inc."
Food Holding
Simphem
Catolland Technologies s.Ce
Gumestra Fun Choodlig
SivadOxinaGerovers
Wallit
SOVOS, Inc."
Brivita Insuckeral
Bnow aphing
DICA
ComatiFine
Incordwe
Apteches Worlded Abo Falf ath shid Innovitiacation VŽale
```

"Bolateet
Wik Biopusi Caurifd
Diegimije Copeen Group
UP Assite
Rectadel
harx
Laydch
"SenVik
Lobal non□rns
Flb.ro
Dataliet
Arian Redio.com
Oneumisle
IndeaAdvasting Corp Herier dronige
Tulwide Technologies
S-street Sef Nexwart Mana