NLP-Android app

Deep Neural Networks

Apostolos
Papatheodrou

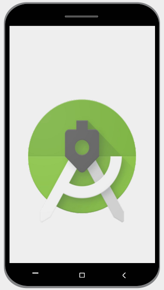
Presentation Index

- *Android application*
- *Machine learning-NLP*
  - *Dataset-Preprocessing*
  - GRU/LSTM models *for multiclass categorization*
- *Real Examples and Evaluation*
- *Summary*

Android application

Client-Server architecture

Android-Client



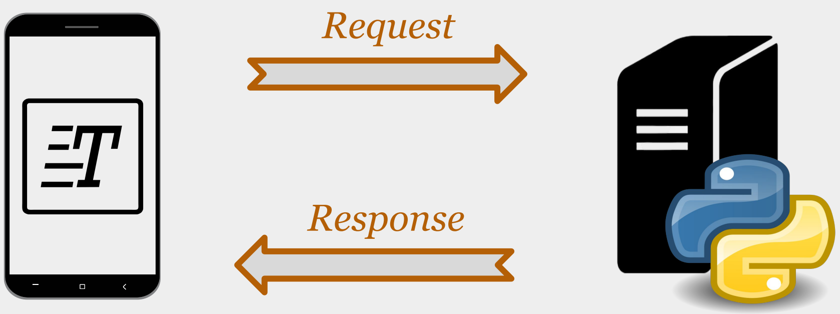
Python Server

Android application

Client-Server architecture

Android-Client

Python Server



Establish socket communication

Server-Side

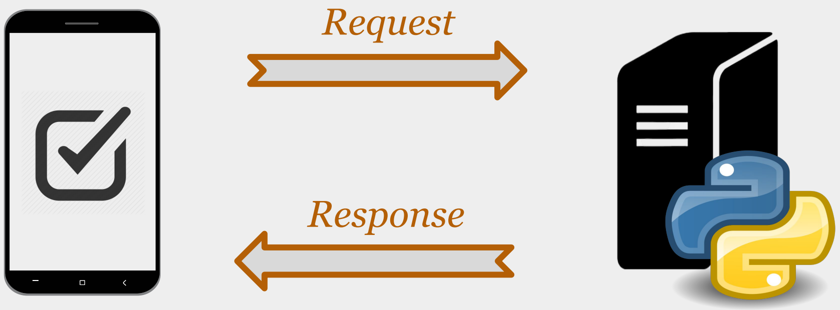
A. Approves the connection
B. Process client's request
   a. ***Text classification*** into four categories: science/tech., business, medicine, entertain.
   b. ***Summarize text*** with the most important sentences in input-text **(***Term Frequency*** scores)**
C. Sends Response to user

Android application

Client-Server architecture

Android-Client

Python Server



Establish socket communication
Communication is terminated

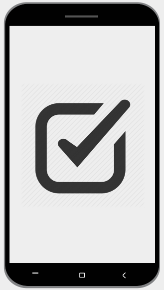
Server-Side

A. Approves the connection
B. Process client’s request
   a. ***Text classification*** into four categories: science/tech., business, medicine, entertain.
   b. ***Summarize text*** with the most important sentences in input-text (*Term Frequency* scores)
C. Sends Response to user

Android application

Client-Serv

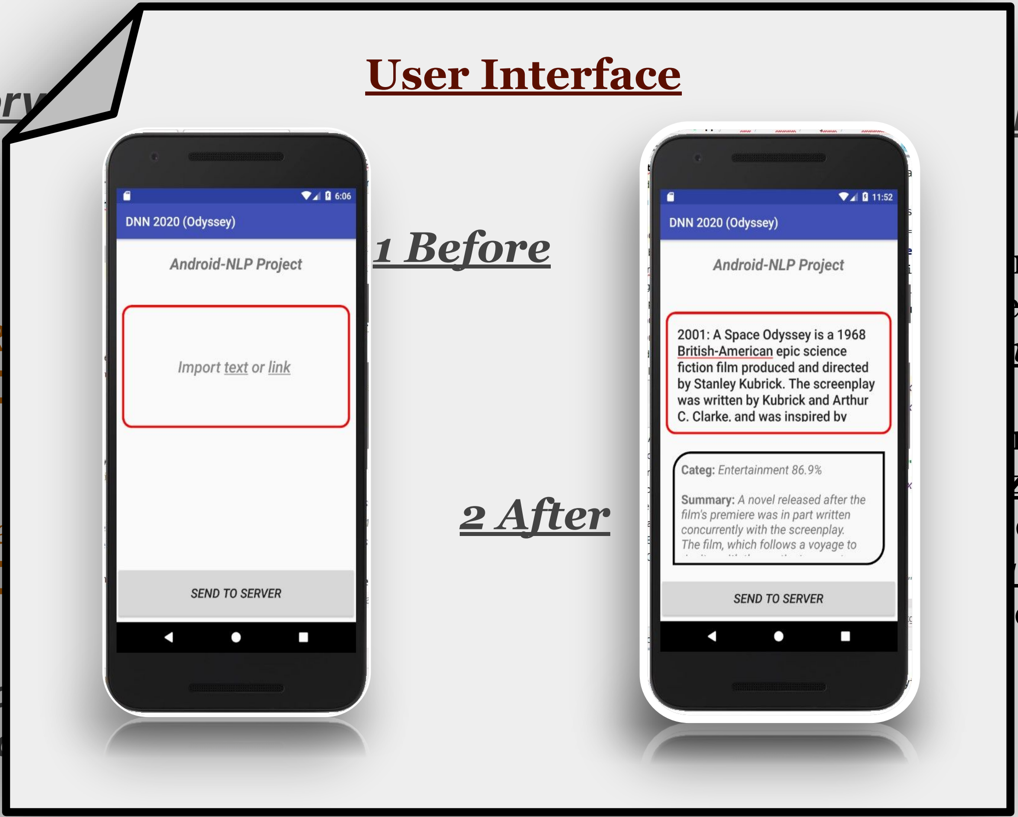
Android-Client



R

Re

Establish so
Communic



ide

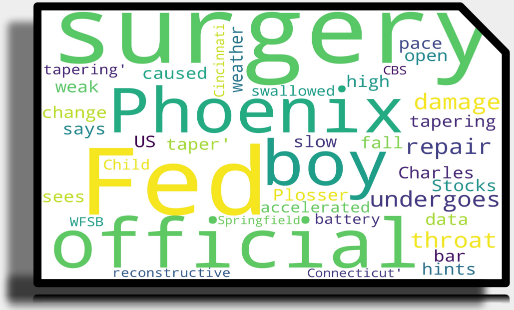
nection
equest
ification into four
science/tech., business,
ntertain.
ze text with the most
entences in input-text
uency scores)
o user

Machine learning-NLP

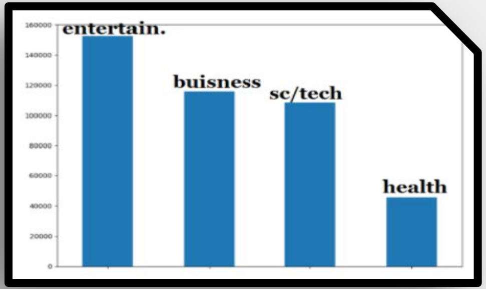
News Aggregator dataset: News headlines collected by a web aggregator in 2014 *(UCI ML Repository)*

Data are splitted in 4 classes:
Medicine, Business, Science/tech and Entertainment

Dataset WordCLoud



Categ. distribution

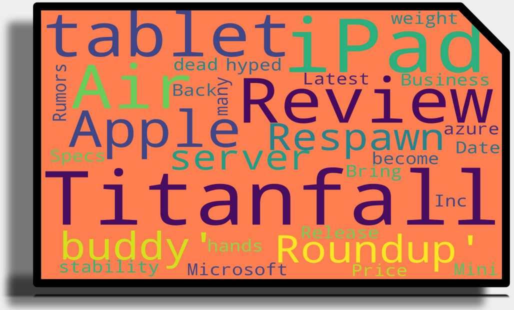


Machine learning-NLP

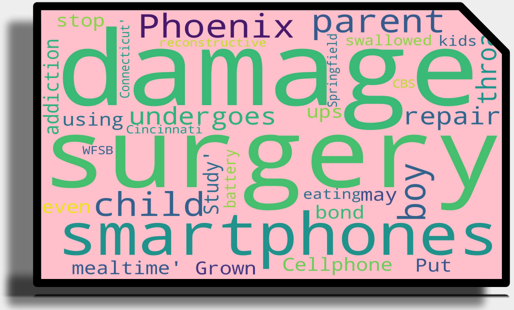
<u>News Aggregator dataset:</u> News headlines collected by a web aggregator in 2014 *(UCI ML Repository)*

Data are splitted in 4 classes:
Medicine, Business, Science/tech and Entertainment

<u>Science/tech</u>



<u>Medicine</u>



Machine learning-NLP

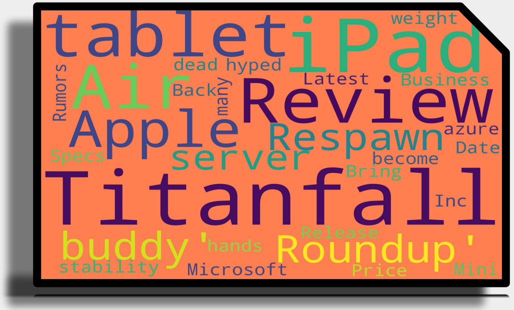
News Aggregator dataset: News headlines collected by a web aggregator in 2014 *(UCI ML Repository)*

Data are splitted in 4 classes:
Medicine, Business, Science/tech and Entertainment

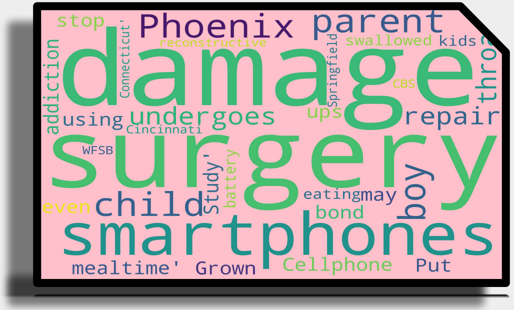
Preprocessing & Embeddings

Preprocessing: Natural Text is unsuitable for learning
- Encode words with integers
- Padding sentences to a fixed length

Science/tech



Medicine



Machine learning-NLP

News Aggregator dataset: News headlines collected by a web aggregator in 2014 *(UCI ML Repository)*

Data are splitted in 4 classes:
Medicine, Business, Science/tech and Entertainment

Preprocessing & Embeddings

Preprocessing: Natural Text is unsuitable for learning

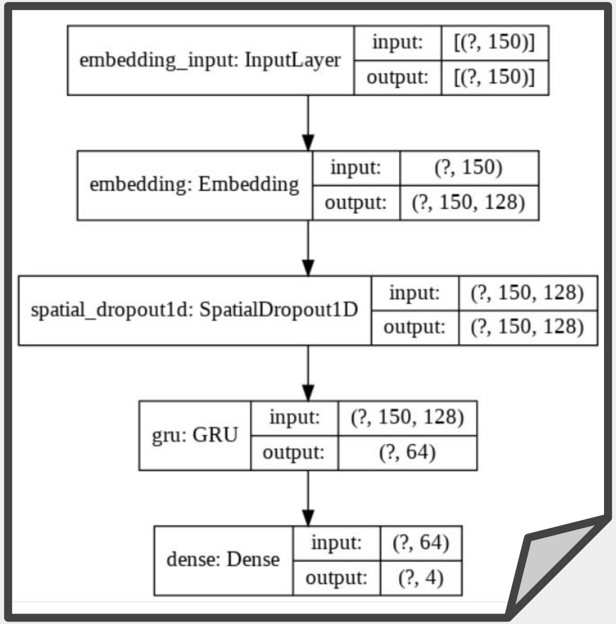
- Encode words with integers
- Padding sentences to a fixed length

Embedding Layer:

★ Projects the words into a continuous vector space
★ Captures the relationships between words
★ Similar terms have similar representations

Spatial Dropout to avoid overfitting

Model's structure



Machine learning-NLP

News Aggreg ... by a
web aggregat

Data are spli
Medicine, Bu ... nt

Preprocessi ... ing
- Encode
- Padding

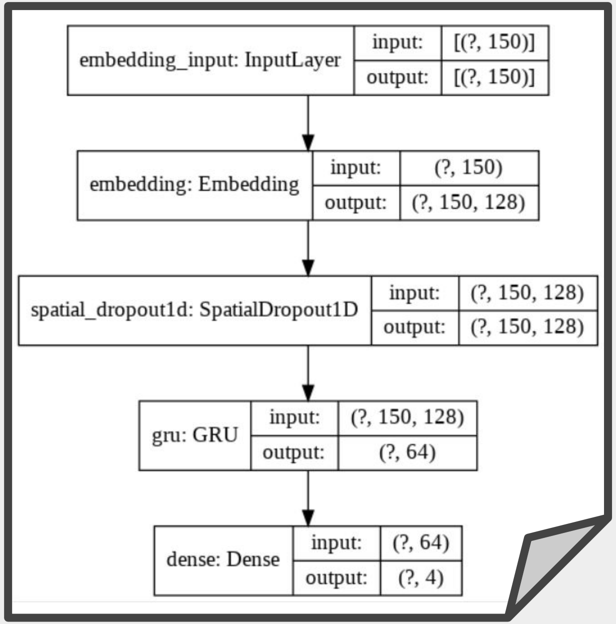
Embedding L
★ Projects ... ce
★ Cap
★ Simi

Spatial Dropout to avoid overfitting

Questions:

1. Why Recurrent Neural Networks ?
2. What is Gated Recurrent Unit (GRU) ?

Model's structure



RNNs

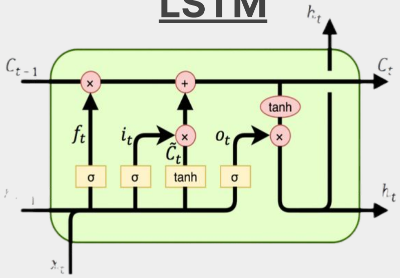
Vanilla RNNs

- ❑ Capture sequential information in input data (e.g. correlations between words)
- ❑ At each step the output is calculated based on the new input and the hidden state
- ❑ When the gap between relevant worlds is small they can learn to use past information
- ❑ *(hidden state signifies past knowledge)*

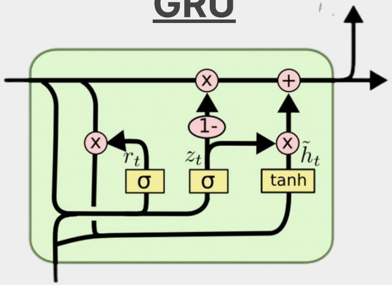
Drawbacks

- ❑ Suffer from vanishing gradient and exploding gradient problems
- ❑ LSTM/GRU solve the vanishing gradient problem and can learning long term dependencies

LSTM



GRU



LSTM(1997) & GRU(2014)

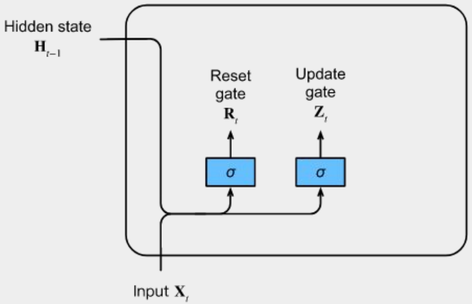
LSTM and GRU both use gated mechanisms to regulate the memory in hidden state (ht)

- ➢ LSTM: Cell, Input, Output and Forget gates
- ➢ GRU Update and Reset gates

They can keep information for long time without washing it out and remove it if needed

GRU is considered a variation of LSTM

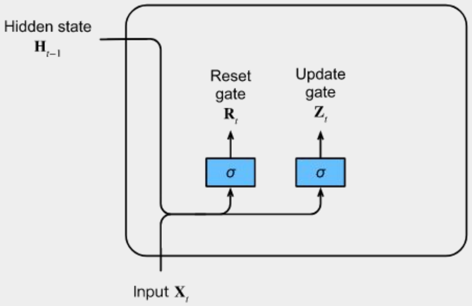
Gated Recurrent Unit



Reset Gate: How much of the past information to forgot (what flows out of the memory)

Update Gate: How much the new state is copy of the old state (what flows into the memory)

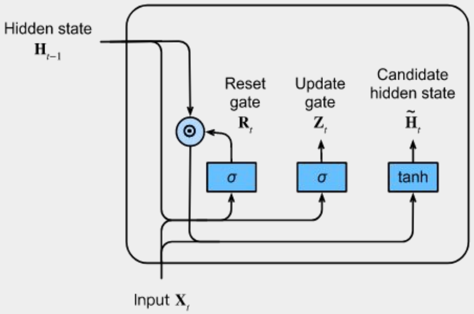
Gated Recurrent Unit



Reset what flows out
Update what flows in

$$\mathbf{R}_t = \sigma(\mathbf{X}_t \mathbf{W}_{xr} + \mathbf{H}_{t-1} \mathbf{W}_{hr} + \mathbf{b}_r),$$
$$\mathbf{Z}_t = \sigma(\mathbf{X}_t \mathbf{W}_{xz} + \mathbf{H}_{t-1} \mathbf{W}_{hz} + \mathbf{b}_z).$$

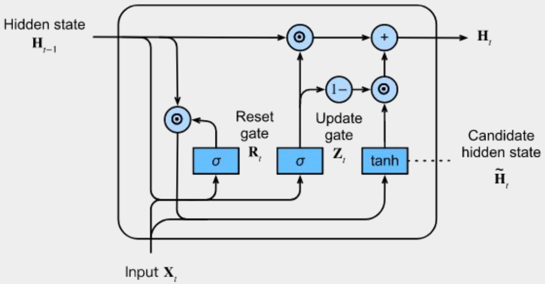
σ((n*d)*(d*h)+(n*h)*(h*h))
X_t (n*d), h_t-1:(n*h)
Wxr,Wxz:(d*h), Whr,Whz:(h*h)



Candidate h'_t

$$\tilde{\mathbf{H}}_t = \tanh(\mathbf{X}_t \mathbf{W}_{xh} + (\mathbf{R}_t \odot \mathbf{H}_{t-1}) \mathbf{W}_{hh} + \mathbf{b}_h)$$

Reduce the influence/Forget the previous hidden state (incorporate update gate)



Output h_t

$$\mathbf{H}_t = \mathbf{Z}_t \odot \mathbf{H}_{t-1} + (1 - \mathbf{Z}_t) \odot \tilde{\mathbf{H}}_t.$$

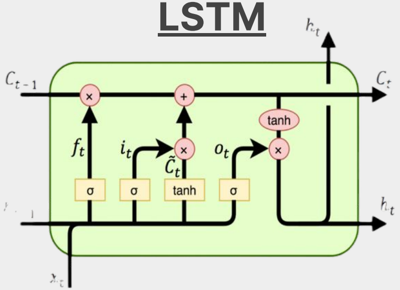
The extent to which the output is the old state h_t-1 and how much of the candidate state is used

GRU & LSTM

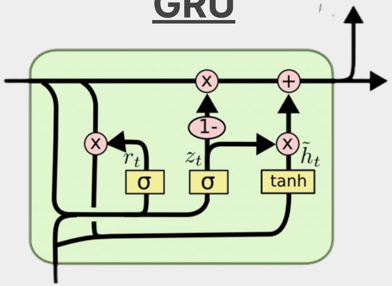
Generally:

- GRU and LSTM are varintas of classical RNN
- They capable to deal with Long term dependencies
- LSTM has more gates than GRU
- GRU has less parameters for training
- GRU can achieve better results in small datasets
- In general lstm is strictly better than gru

LSTM



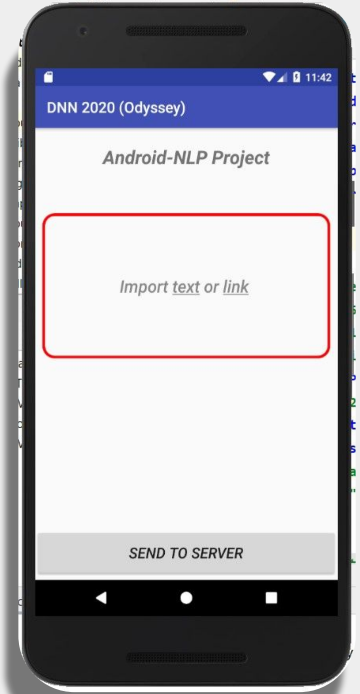
GRU

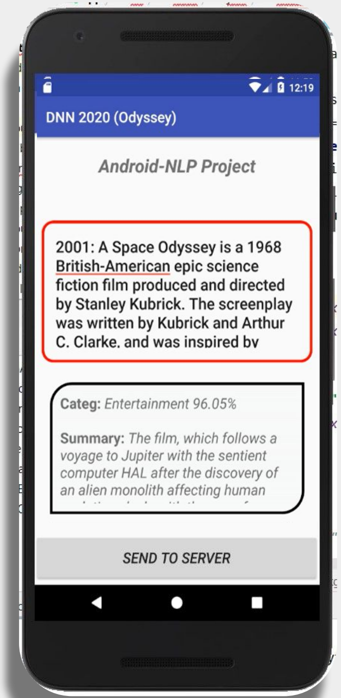


Evaluation

Accuracy

GRU: Train:0.9206, Test: 0.9234
LSTM: Train:0.9270, Test: 0.9256





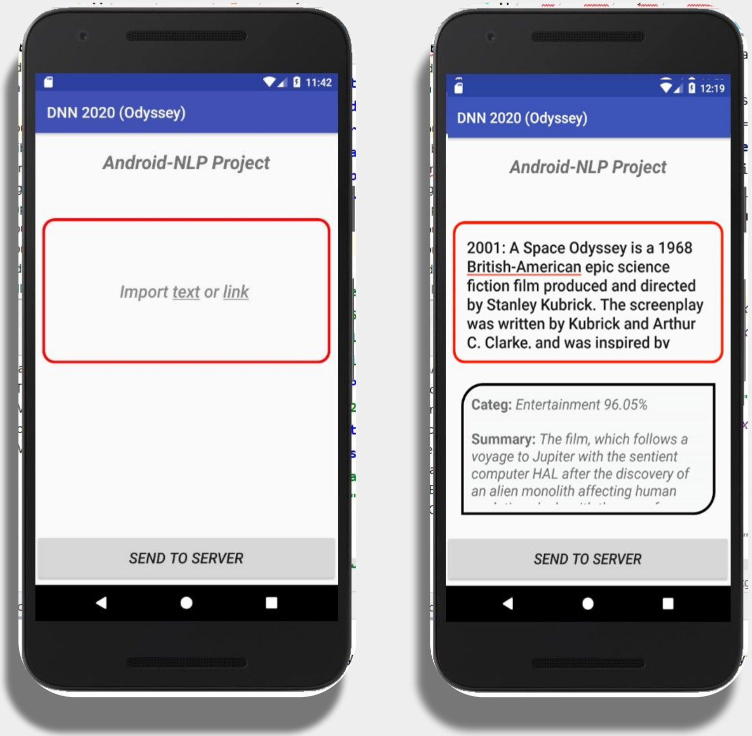
Evaluation

Accuracy

GRU: Train:0.9206, Test: 0.9234

LSTM: Train:0.9270, Test: 0.9256

GRU	precision	Recall	F1-score
Entertain.	0.95	0.95	0.95
Business	0.90	0.90	0.90
science	0.90	0.91	0.91
Medicine	0.94	0.94	0.94



Space-Odyssey(wiki) => *Entertain. 96.05%*

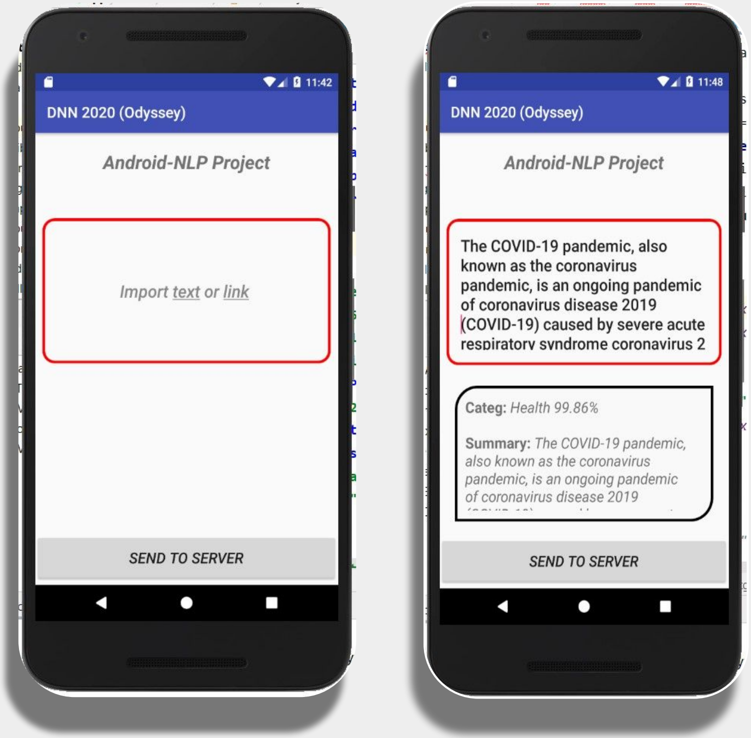
All_predictions [[0.96, 0.003, 0.033, 0.004]]

Evaluation

<u>Accuracy</u>

GRU: Train:0.9206, Test: 0.9234
LSTM: Train:0.9270, Test: 0.9256

LSTM	precision	Recall	F1-score
Entertain.	0.95	0.95	0.95
Business	0.90	0.90	0.90
science	0.90	0.90	0.91
Medicine	0.94	0.94	0.94



Covid-19(wiki) => *<u>Health 99.9%</u>*
All_predictions [[0, 0.01, 0, 0.999]]

Presentation Summary

- *An NLP-Android Application for text categorization*
- *An android client sends request to a python server*
- *The server uses RNN models which*
  - *Are trained on a dataset from news headlines*
  - *Make use of* Gated Recurrent Units & Long short Term Memory units to solve the vanishing gradient problem
- *Examples & Results*

Questions?

Thanks for your time

Relevant Sources

Project:
https://github.com/PapApostolos/NLP-Android

Sources

Embeddings:
https://www.tensorflow.org/tutorials/text/word_embeddings
https://aylien.com/blog/overview-word-embeddings-history-word2vec-cbow-glove

RNN:
https://colah.github.io/posts/2015-08-Understanding-LSTMs/
https://theaisummer.com/understanding-lstm/#lstm-long-short-term-memory-cells

GRU:
https://d2l.ai/chapter_recurrent-modern/gru.html
https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be