Sapienza
Università di Roma

Word-in-Context Disambiguation

NLP Homework 1
July 2021

Olga Sorokoletova, 1937430

Overview

- Given: **Word-in-Context Disambiguation task**
  - **Word-level approach**
  - **Sequence encoding approach**
- Goal: **Obtain the best-performing model (in terms of accuracy)**
  - A priori
  - Exploit potential power of the sequence encoding approach

- Models:
  - **Baseline**
  - **Baseline 2**
  - **2a**
  - **2b**

Baseline

Pre-processing

Pre-processing: Single sentence processing

Given: `'Over 5,000 now hold legal immigrant documents, which, after five years of annual renewal, entitles the` **holder** `to apply for permanent residence.'`

1. **Numbers Removal** — `Over , now hold legal immigrant documents, which, after five years of annual renewal, entitles the holder to apply for permanent residence.`

2. **Punctuation Removal** — `Over  now hold legal immigrant documents which after five years of annual renewal entitles the holder to apply for permanent residence`

3. **Lower Casing** — `over  now hold legal immigrant documents which after five years of annual renewal entitles the holder to apply for permanent residence`

4. **Tokenization** — ['over', 'now', 'hold', 'legal', 'immigrant', 'documents', 'which', 'after', 'five', 'years', 'of', 'annual', 'renewal', 'entitles', 'the', 'holder', 'to', 'apply', 'for', 'permanent', 'residence']

5. **Stop words removal** — ['hold', 'legal', 'immigrant', 'documents', 'five', 'years', 'annual', 'renewal', 'entitles', 'holder', 'apply', 'permanent', 'residence']

Stop words removal problem

Before:

'At the police station he did not make any such claims, but had alleged torture **only** at the district court trials.'

After:

['police', 'station', 'make', 'claims', 'alleged', 'torture', 'district', 'court', 'trials']

Needs to be manually corrected!

Embeddings

GloVe 50d

['hold', 'legal', 'immigrant', 'documents', 'five', 'years', 'annual', 'renewal', 'entitles', 'holder', 'apply', 'permanent', 'residence']

[**embedding**('hold)', **embedding**('legal'), **embedding**('immigrant'),...]

Compute mean

```
[ 0.3866, -0.2908, -0.1011,
0.1910,  0.1187,  0.1503,  0.1034,
0.3662, 0.2403, -0.2460, -0.2139,
-0.4824, -0.3774, -0.4342,
0.5688, -0.1132,...]
```

50d tensor of numbers

Pre-processing: Join

'sentence1': 'This growth is the direct result of the increased number of baccalaureate holders, who form the potential market for higher education.'

'.'

'sentence2': 'Over 5,000 now hold legal immigrant documents, which, after five years of annual renewal, entitles the holder to apply for permanent residence.'

[0.1160, 0.3436, 0.2790,...]

50d tensor of numbers

[1.5164e-01, 3.0177e-01, -1.6763e-01,...]

50d tensor of numbers

[-0.0470, 0.5142, -0.0584,...]

50d tensor of numbers

[0.1160, 0.3436, 0.2790,..., 1.5164e-01, 3.0177e-01, -1.6763e-01,...,-0.0470, 0.5142, -0.0584,...]

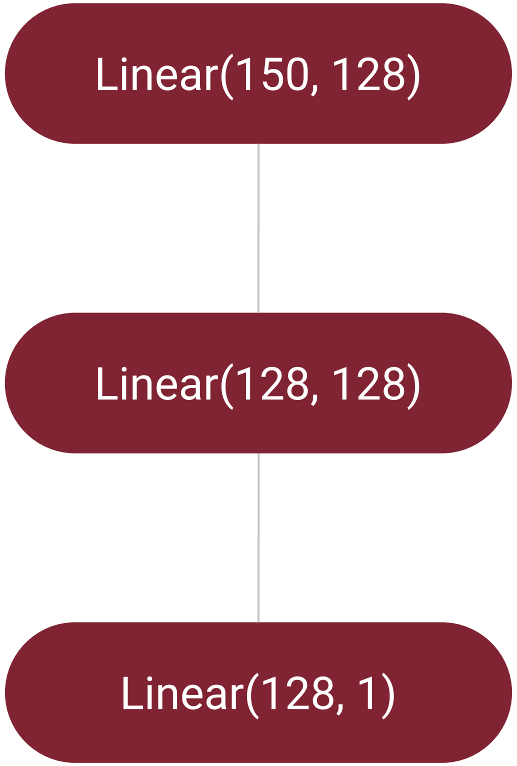
150d tensor of numbers

Baseline

Model

Model

Architecture



Hyper-parameters

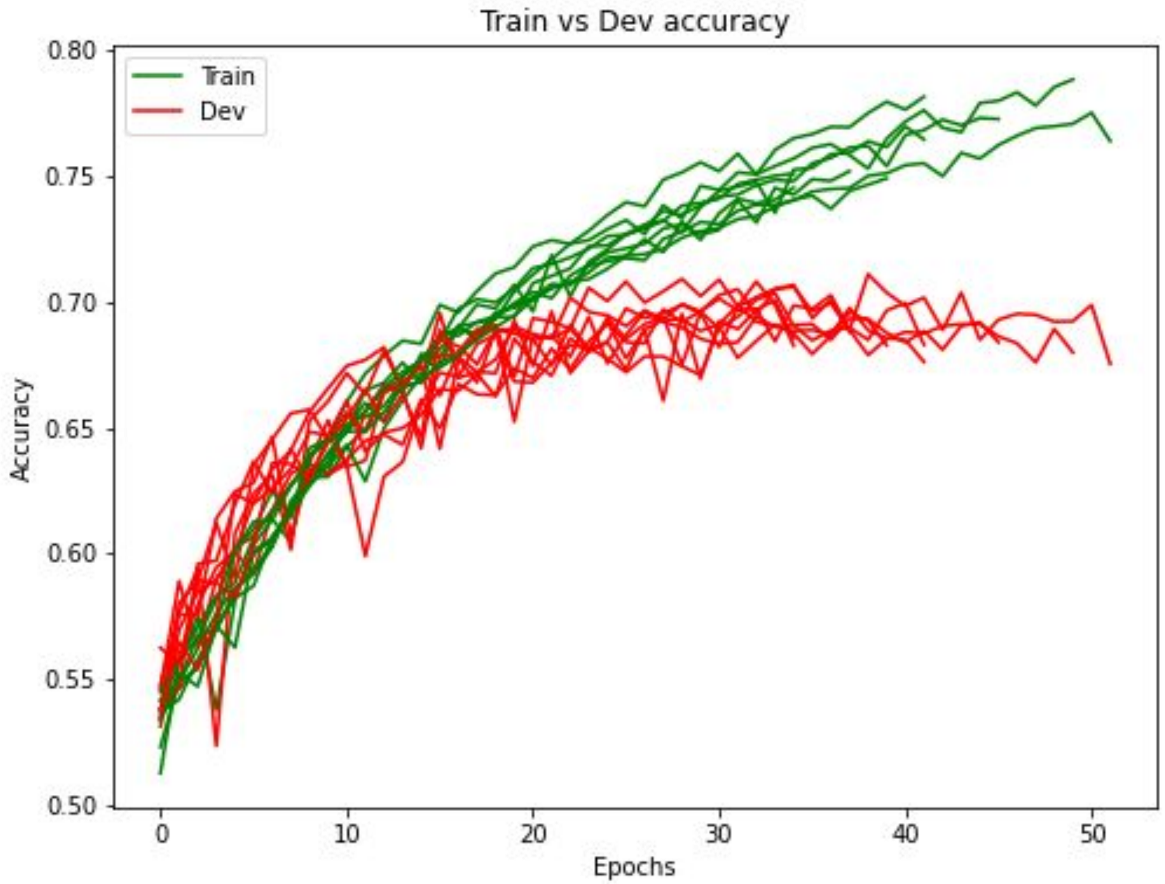
Epochs	50
ES patience	7
ES threshold	0.009
Batch size	64
Embedding dim	50
N features	150
N hidden units	128
N hidden layers	2
Activation	ReLU
Optimizer	Adam
Learning Rate	0.0001

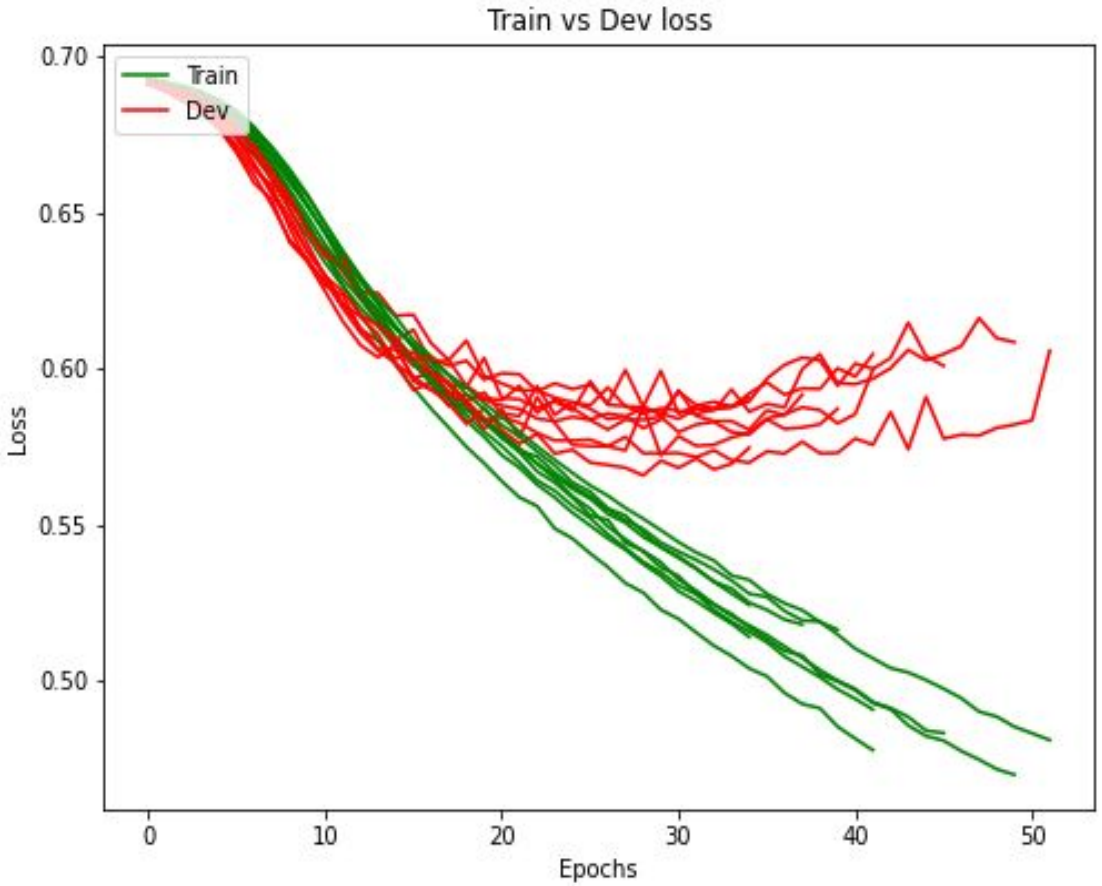
Baseline

Performance

Performance: Training vs Validation Accuracy and Loss

- Best accuracy: **0.7236**
- Problem: **Overfitting about 30-40 epoch (regularization does not help!)**





Over 10 independent runs

Baseline 2

Pre-processing

Pre-processing: Single sentence processing

Given: `'It will place as many demands on our material resources as on our intellectual capabilities.'`

1. Numbers Removal
2. Punctuation Removal
3. Lower Casing
4. Tokenization
5. Stop words removal

['place', 'many', **'demands'**, 'material', **'resources'**, 'intellectual', **'capabilities'**]

6. Lemmatization

['place', 'many', **'demand'**, 'material', **'resource'**, 'intellectual', **'capability'**]

Pre-processing: Join

'**sentence1**': 'This growth is the direct result of the increased number of baccalaureate holders, who form the potential market for higher education.'

'.'

'**sentence2**': 'Over 5,000 now hold legal immigrant documents, which, after five years of annual renewal, entitles the holder to apply for permanent residence.'

['growth', 'direct', 'result', 'increased', 'number', 'baccalaureate', 'holder', 'form', 'potential', 'market', 'higher', 'education']

['.']

['hold', 'legal', 'immigrant', 'document', 'five', 'year', 'annual', 'renewal', 'entitles', 'holder', 'apply', 'permanent', 'residence']

['growth', 'direct', 'result', 'increased', 'number', 'baccalaureate', 'holder', 'form', 'potential', 'market', 'higher', 'education',

'.',

'hold', 'legal', 'immigrant', 'document', 'five', 'year', 'annual', 'renewal', 'entitles', 'holder', 'apply', 'permanent', 'residence']

Pre-processing: Prepare input for RNN

['growth', 'direct', 'result', 'increased', 'number', 'baccalaureate', 'holder', 'form', 'potential', 'market', 'higher', 'education', '.', 'hold', 'legal', 'immigrant', 'document', 'five', 'year', 'annual', 'renewal', 'entitles', 'holder', 'apply', 'permanent', 'residence']

Indexed vocabulary with 2 special indices:

- **0 - for padding token**
- **1 - for Out-of-Vocabulary**

$$[554, 1496, 714, 1043, 225, 31822, 6101, 685, 1158, 213, 611, 633, 4, 804, 832, 5660, 2883, 176, 64, 942, 9239, 53808, 6101, 3517, 2275, 3700, \underbrace{0, 0, \ldots, 0}]$$

Padding of a sequence with ML - L zeros:

- **ML - max length of a sequence in the batch**
- **L - length of the current sequence**

GloVe 50d

(random for 0 and 1)

[**embedding**(554), **embedding**(1496),…, **embedding**(0)]

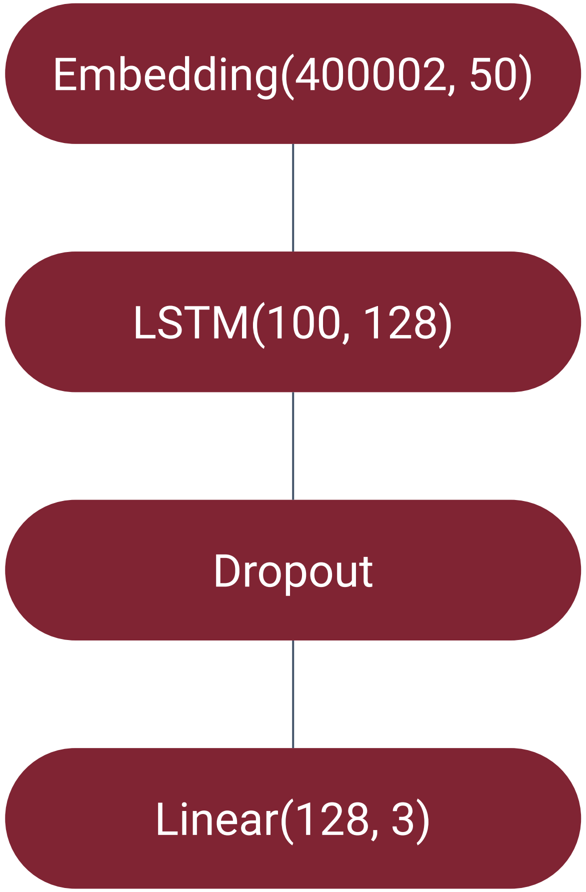
50 x ML tensor of numbers

Baseline 2

Model

Model

Architecture



Hyper-parameters

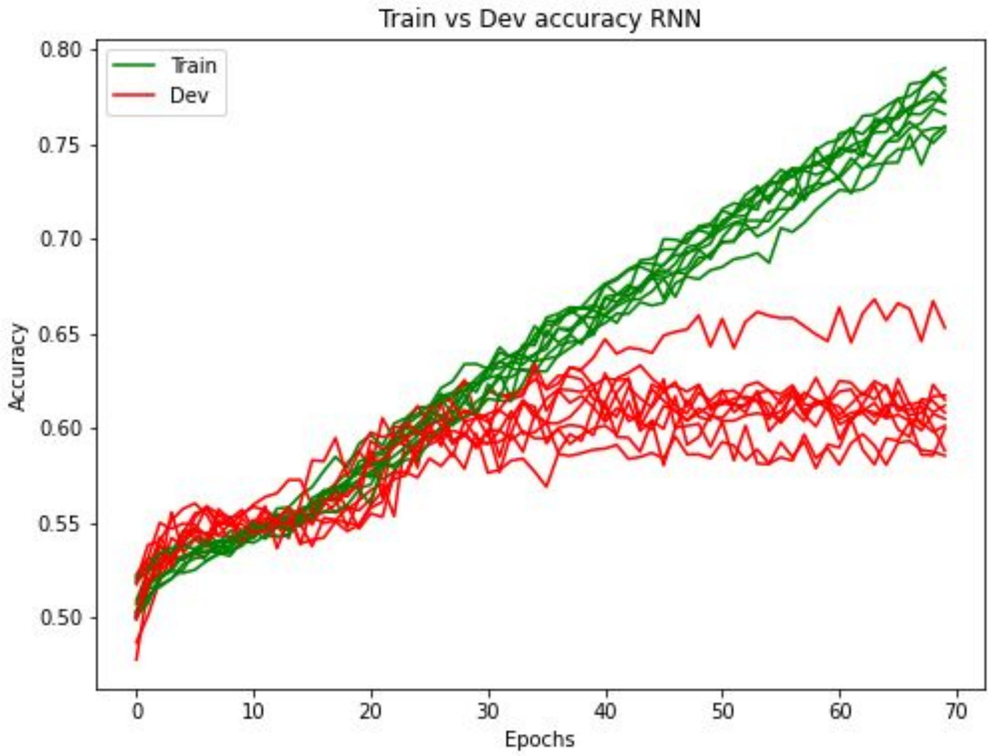
Epochs	70
ES patience	7
ES threshold	0.01
Batch size	256
Embedding dim	50
N features	50
N hidden units	128
N LSTM cells	1
Activation	ReLU
Optimizer	Adam
Learning Rate	0.0001
Decay Rate	0.000001
Dropout Rate	0.0

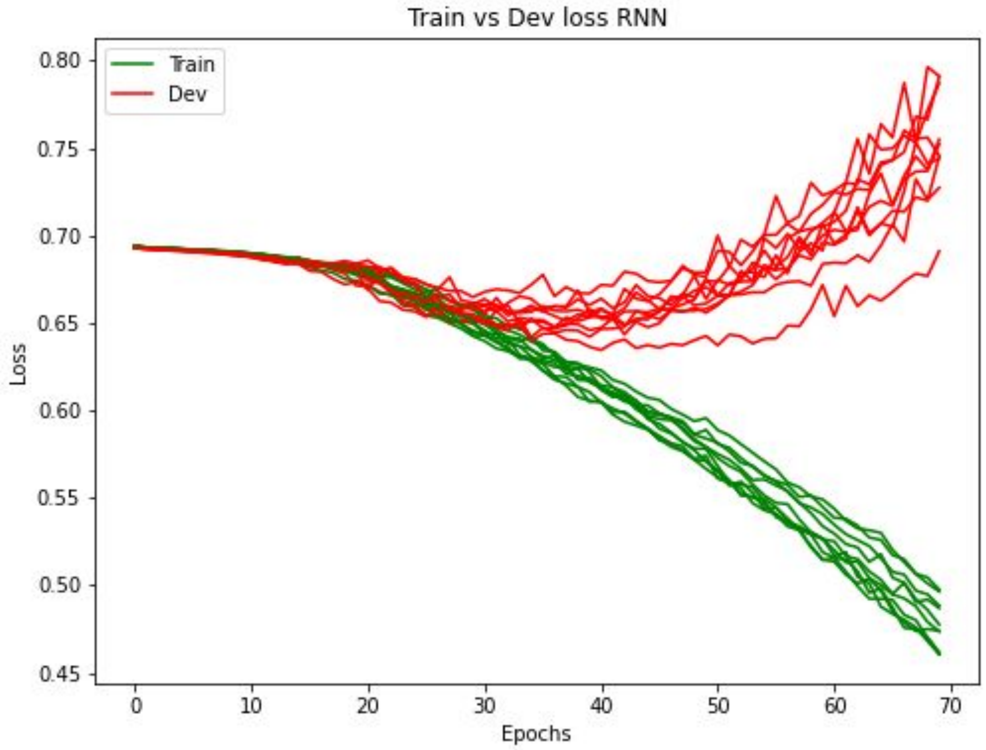
Baseline 2

Performance

Performance: Training vs Validation Accuracy and Loss

- Best accuracy: **0.6681 (vs 0.7236 we had before)**
- Problem: **Overfitting about 50 epoch**

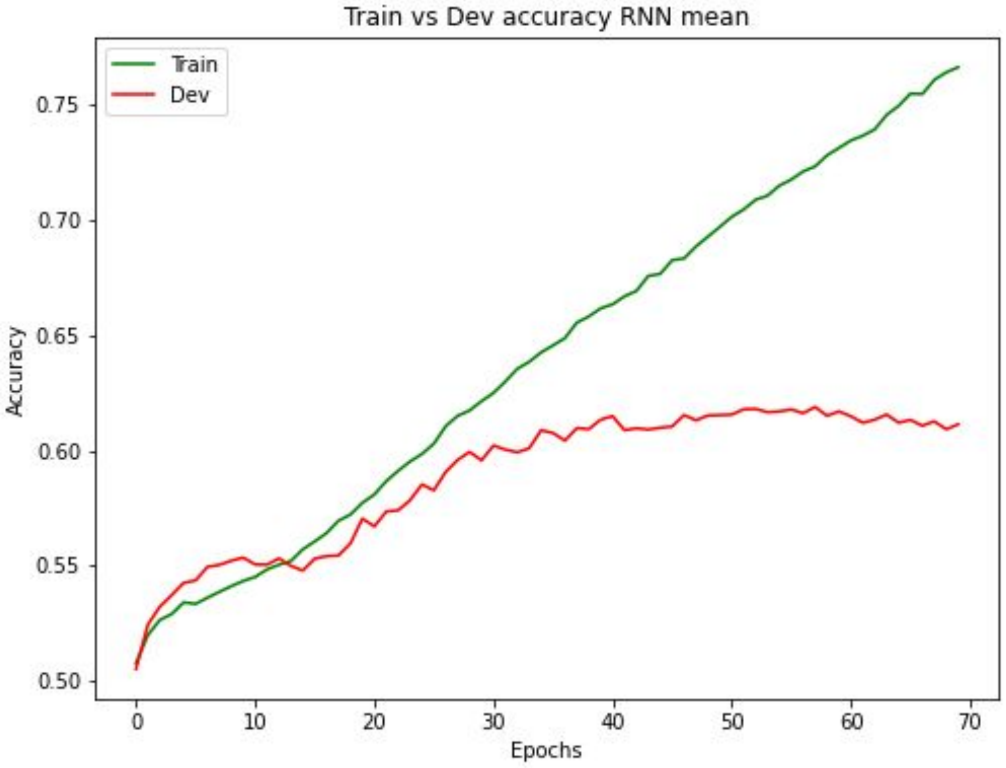


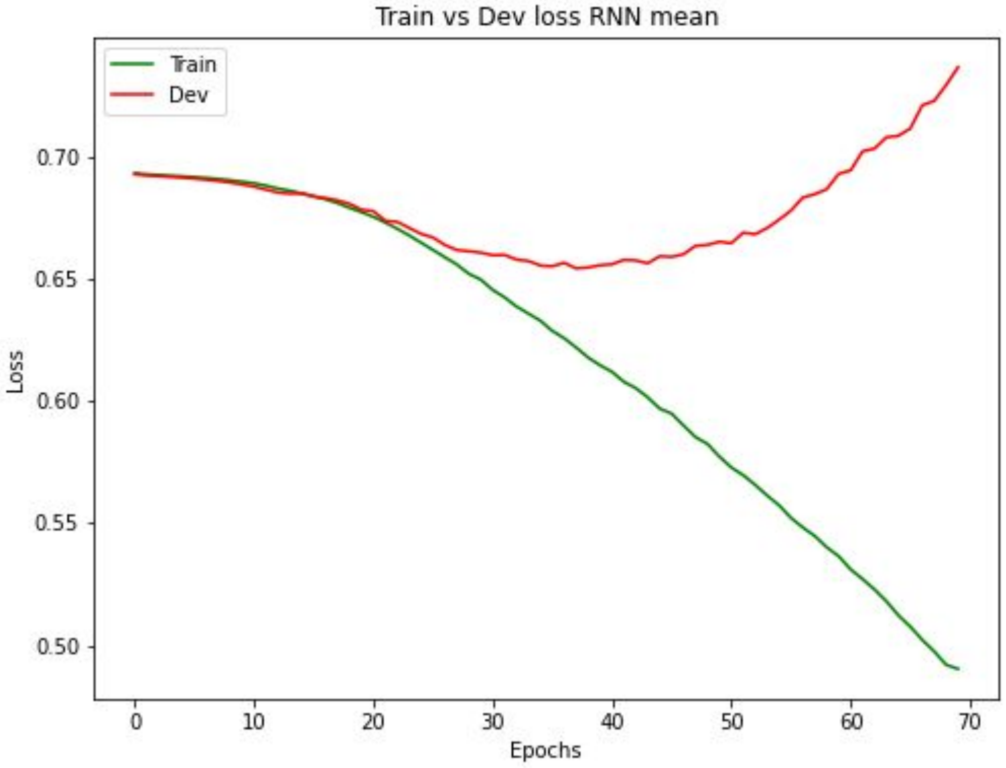


Over 10 independent runs

Performance: Averaged graphs

- Accuracy: got stabilized about **0.60 (vs 0.66 the best accuracy)**
- Problem: **overfitting, the best performance achieved by lucky initialization**

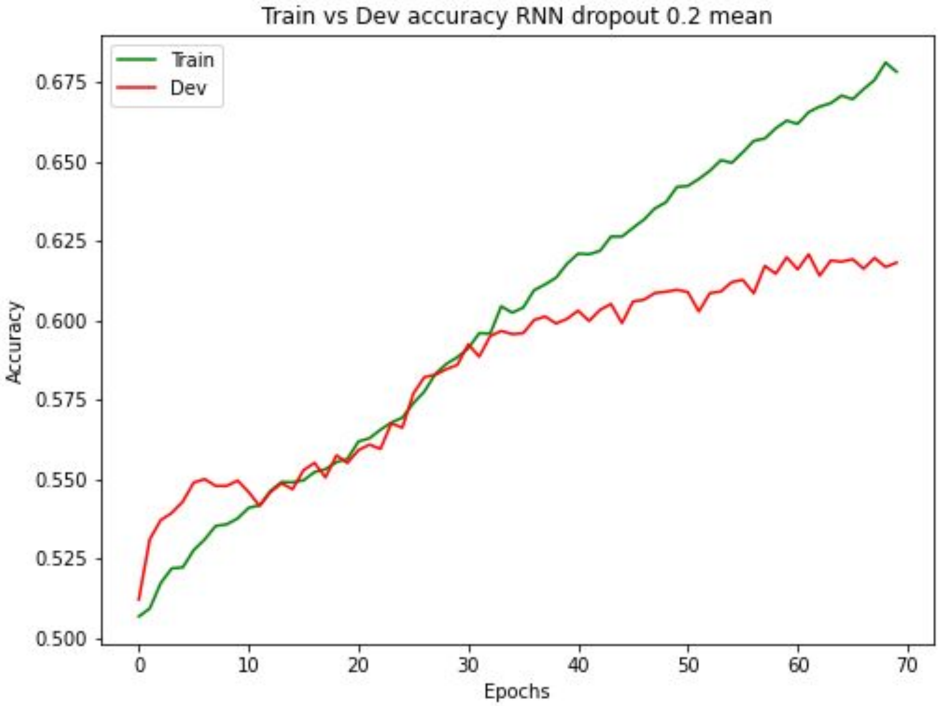


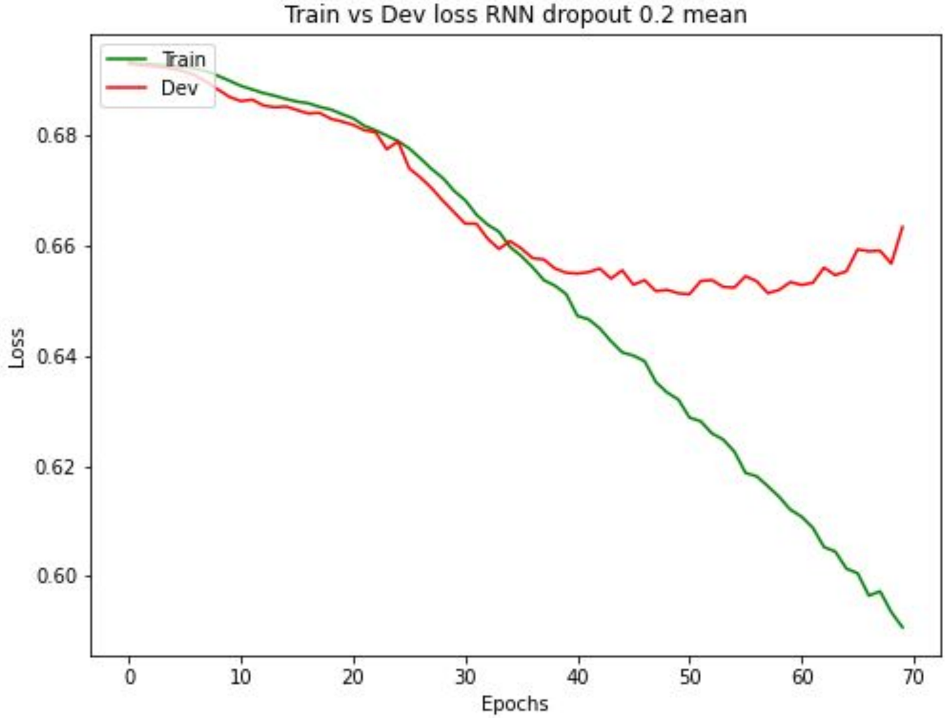


Averaged over 10 independent runs

Performance: Add dropout

- Accuracy: got stabilized about **0.62 (vs 0.60 without dropout)**
- Problem: **overfitting, but less obvious**

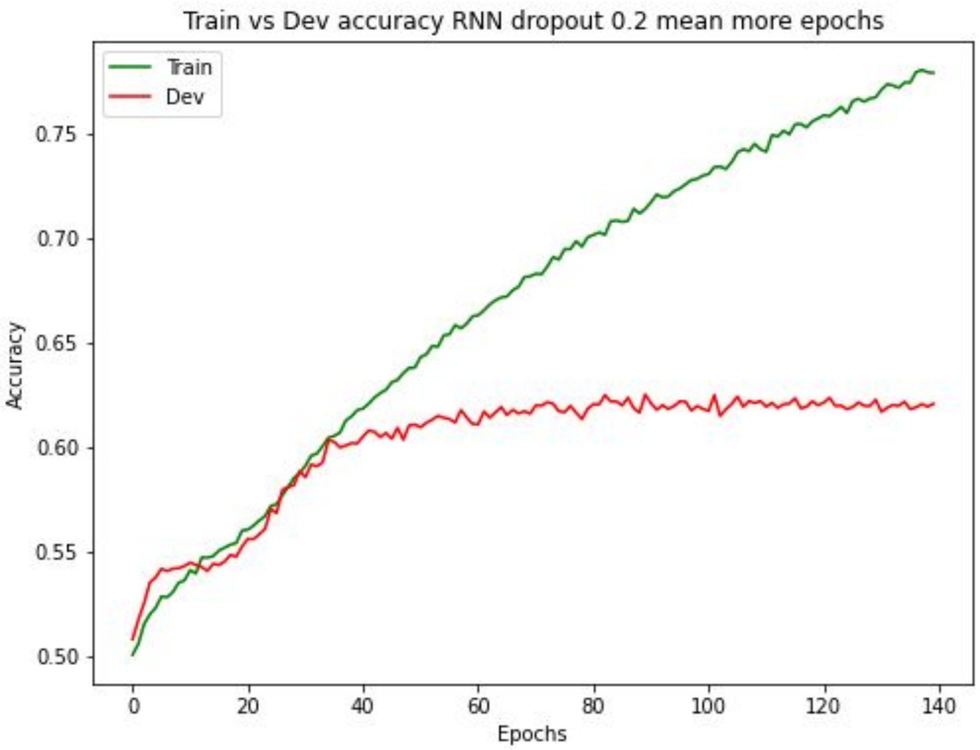


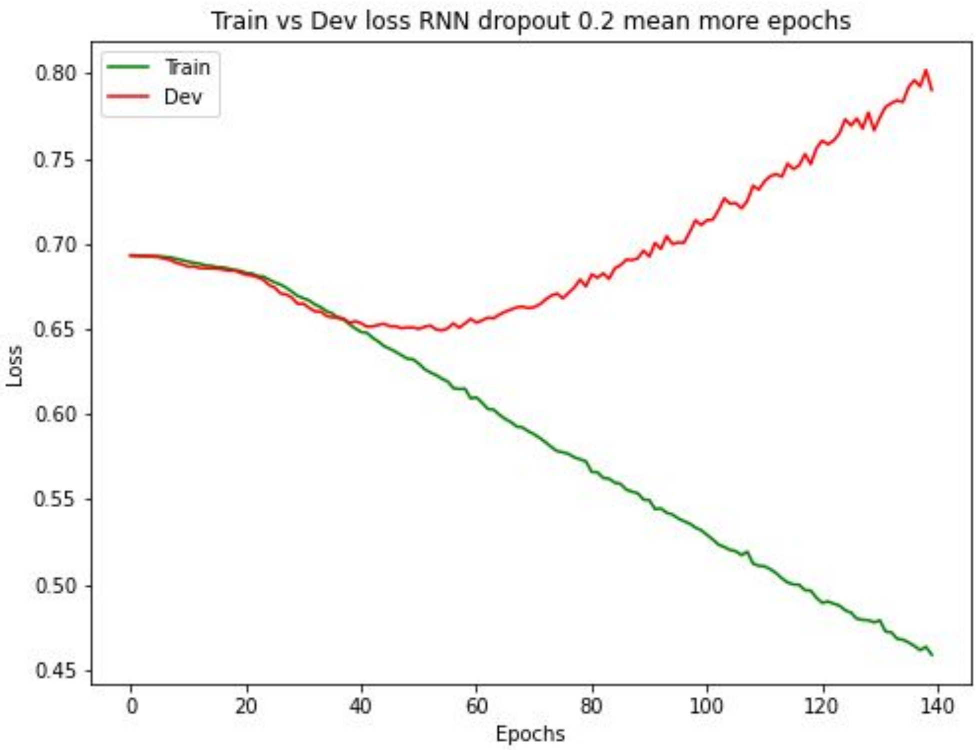


Averaged over 10 independent runs, p = 0.2, 2 LSTM layers, dropout applied after embedding layer, between 2 LSTMs and after them

Performance: More epochs

- Accuracy: got stabilized about **0.62 (vs 0.62)**
- Problem: **overfitting, now it is obvious**

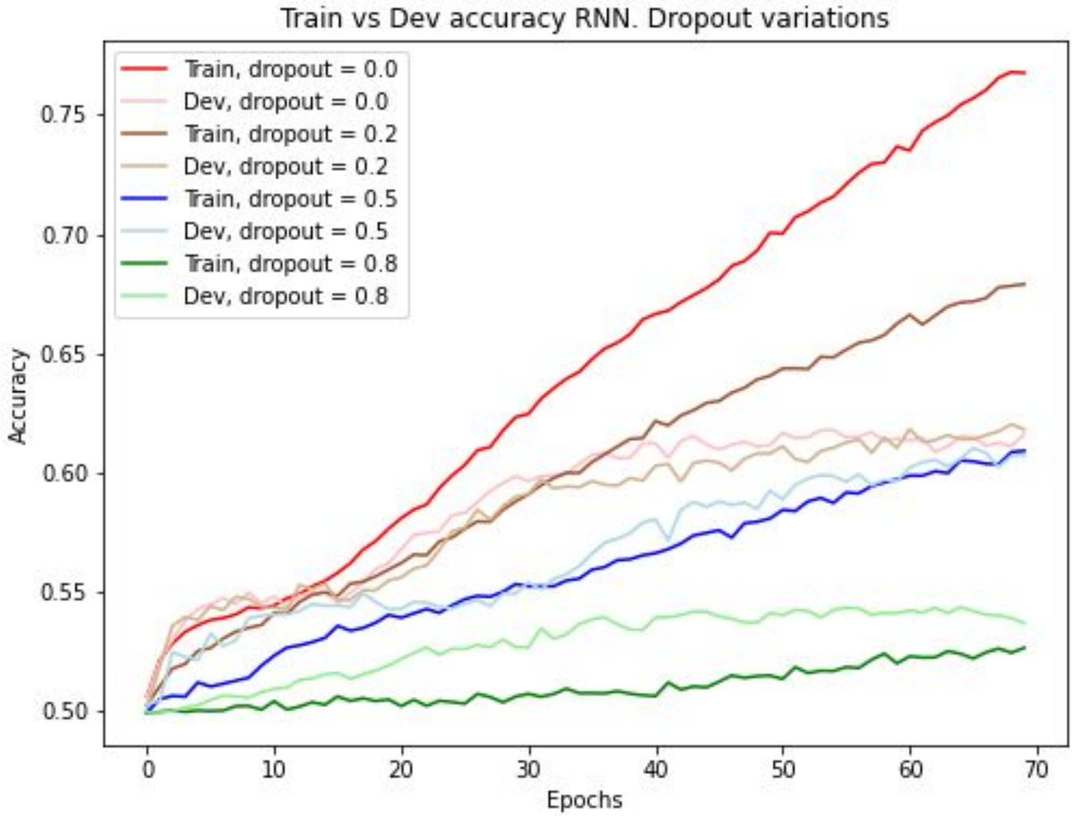




Averaged over 10 independent runs, p = 0.2, 2 LSTM layers, dropout applied after embedding layer, between 2 LSTMs and after them, 140 epochs of training

Performance: Dropout variations

- Accuracy: got stabilized about **0.62 or less**
- Problem: **overfitting, and dropout variation does not help**



Averaged over 10 independent runs

2a

Pre-processing and Model

Pre-processing and Model: differences with Baseline 2

Pre-processing

Need to keep index of the target word in the sequence of indices (additionally to an index of the last not padding token)

Model

Extract two sequence encodings: corresponding to the representation of a whole sentence and corresponding to the target word

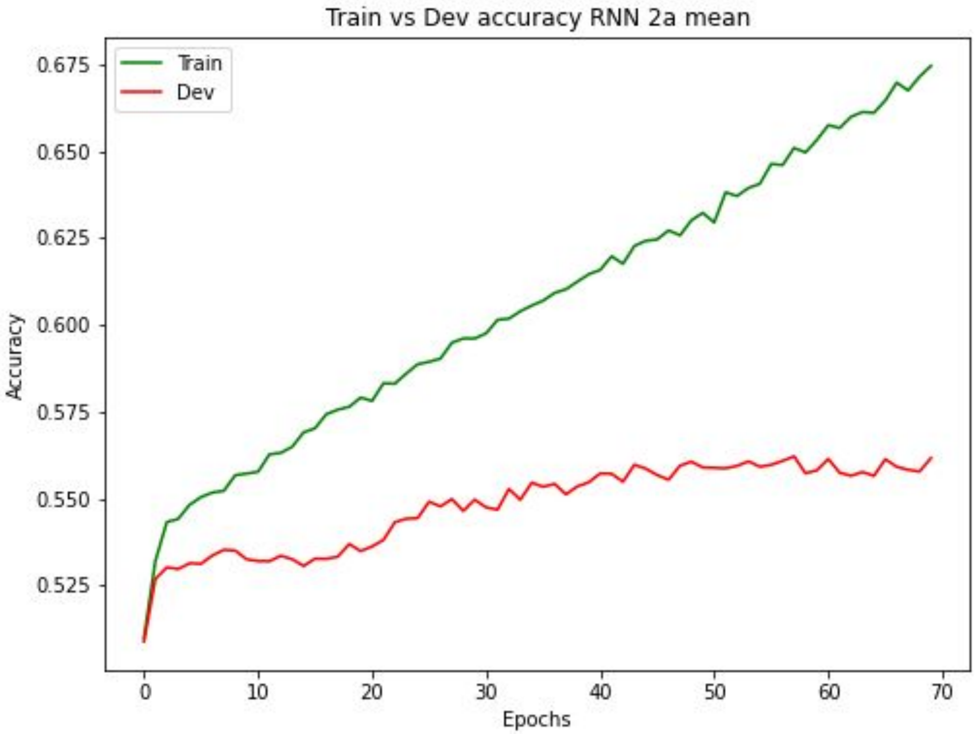
Double-labeling

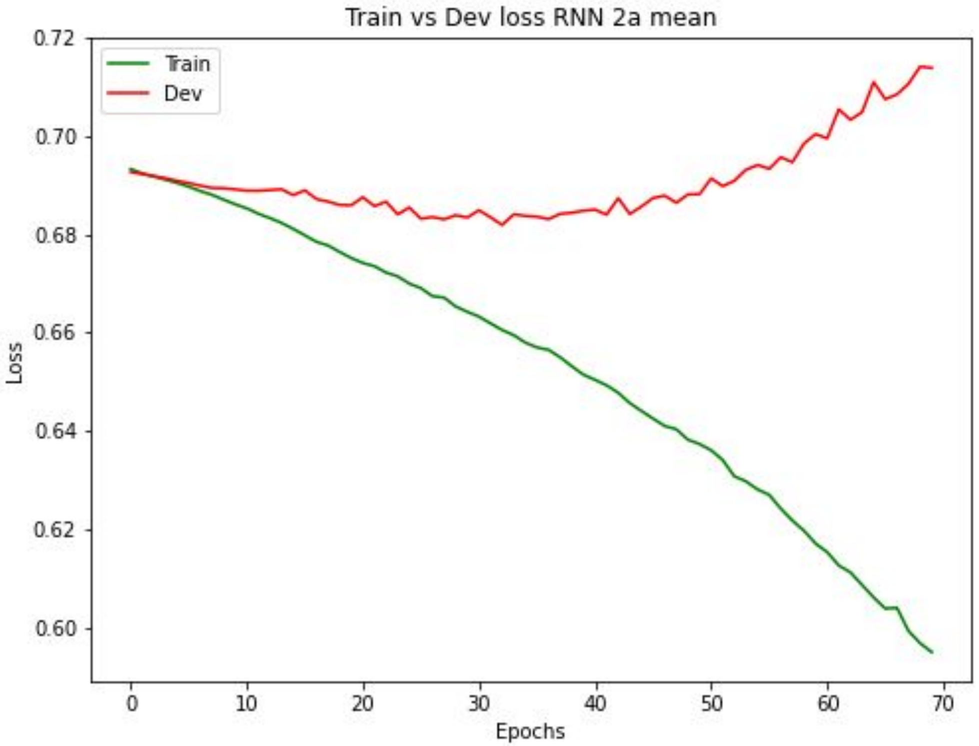
2a

Performance

Performance: Training vs Validation Accuracy and Loss

- Best accuracy: **0.5776 (vs 0.6681 we had for Baseline 2)**
- Problem: **improvement is not achieved**





Averaged over 10 independent runs

2b

Idea

2b

The underlying idea of this model was to perform binary classification over the following representation of the training data:

Feature 1	Feature 2	Feature 3	Feature 4	Label
whole sequence encoding for the sentence 1	sequence encoding corresponding to the target word in the sentence 1	whole sequence encoding for the sentence 2	sequence encoding corresponding to the target word in the sentence 2	gt label

- Best accuracy: **0.5773 - comparable with 2a, but not comparable with Baseline 2**

Conclusion

- Even if sequence encoding approach is potentially more powerful, sometimes **simpler approach can be better** performing;

- Handling **overfitting** is a challenging task, and in our case standard regularization techniques did not help. Therefore, some more sophisticated approaches either to it or to the model architecture design/way of pre-processing are needed;

- However, the best performing model achieved quite **decent performance** by means of common NLP practices for the pre-processing.

SAPIENZA
UNIVERSITÀ DI ROMA

Aspect-Based Sentiment Analysis

NLP Homework 2

July 2021

Olga Sorokoletova, 1937430

Overview

- Given: **Aspect-Based Sentiment Analysis task**
    - **Aspect Extraction**
    - **Aspect Sentiment**
    - **Aspect Extraction + Aspect Sentiment**
    - **Category Extraction**
    - **Category Evaluation**
    - **Category Extraction + Category Sentiment**
- Goal: **Obtain the best-performing model (in terms of macro-F1)**

Models (denoted with just letters are transformer-based):

- **Naive**
- **a**
- **b**
- **c**
- **d**
- **a -> b**
- **a + b**
- **c -> d**
- **c + d**

Naive (just word embeddings)

Pre-processing

Pre-processing

1. First input sentence is tokenized and each token maps to ground truth labels as follows:

1 if token ∈ gt term
0 otherwise

Before:

'The **selection** changes frequently but the **basic dishes** are always available.'

After 1:

['The: 0', 'selection: **1**', 'changes: 0', 'frequently: 0', 'but: 0', 'the: 0', 'basic: **1**', 'dishes: **1**', 'are: 0', 'always: 0', 'available: 0']

2. Then **indexed vocabulary** (with <UNK> and <PAD> tokens indexed as 1 and 0) and **indexed label vocabulary** (with 3 elements: 2 labels and <PAD> token indexed as 2) are created:

Vocab: ['A:2', 'hearty:1', 'two:3',..., 'subwoofer:3553', 'scary:3552']

Label vocab: ['<pad>:2', '0:0', '1:1']

After 2:

[('The', 36), ('selection', 76), ('changes', 1), ('frequently', 77), ('but', 78), ('the', 9), ('basic', 79), ('dishes', 80), ('are', 81), ('always', 82), ('available', 83), ('None', 0), ('None', 0),...,('None', 0)]

[0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 2, 2, …, 2]

SAPIENZA
UNIVERSITÀ DI ROMA

Embedding
s

Two ways of creating indexed vocabulary are applied:

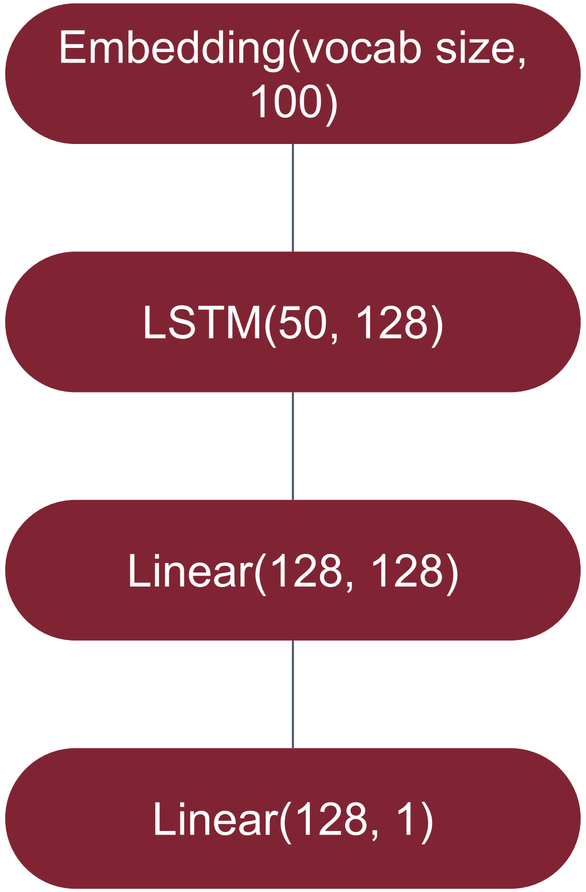
- Based on the dataset;
- Downloading **GloVe 100d** embeddings, so they could be further applied as pre-trained in the network layer.

Naive

Model

Model

Architecture



Hyper-parameters

Epochs	100
Batch size	128
Embedding dim	100
Window Size	100
Window Shift	50
N hidden units	128
N LSTM cells	1
Optimizer	Adam
Learning Rate	0.0001
Dropout Rate	0.0

Naive

Post-processing and Performance

Post-processing: collect predicted tokens to the multi-token terms

```
{'targets': [[[11, 29], 'wines by the glass', 'negative']],
 'text': 'Not enough wines by the glass either.'}
```

Ground truth tokens	Input tokens	Gold labels	Predicted labels	Predicted tokens
	Not	0	0	
	enough	0	0	
wines	wines	1	1	wines
by	by	1	1	by
the	the	1	0	
glass	glass	1	1	glass
	either	0	0	

```
{'wines by', 'glass'}
```

- Best F1: **0.67**

BERT-models

Pre-processing

Pre-processing

- Interpret as a **sentence-pair classification** task:

<CLS> Sentence 1 **<SEP>** Sentence 2 **<SEP>** Label

- Apply BERT tokenizer form pre-trained for the corresponding BERT model:

[**ind(<CLS>)**,
indices(tokenized(sentence 1)), **ind(<SEP>)**,
indices(tokenized(sentence 2)), **ind(<SEP>)**,
indices(padding)]

\+ **attention mask**

[**1**,
1,...,1, **1**,
1,...,1, **1**,
0, …,0]

to Embedding layers

Given

```
{'categories': [['food', 'neutral']],
 'targets': [[[4, 13], 'selection', 'neutral'],
  [[41, 53], 'basic dishes', 'neutral']],
 'text': 'The selection changes frequently but the basic dishes are always available.'}
```

Aspect Extraction

Pre-processing

Binary classification

Sentence (Sentence 1)	Aspect (Sentence 2)	Label
The selection changes frequently but the basic...	The	unrelated
The selection changes frequently but the basic...	selection	**related**
The selection changes frequently but the basic...	changes	unrelated
The selection changes frequently but the basic...	frequently	unrelated
The selection changes frequently but the basic...	but	unrelated
The selection changes frequently but the basic...	the	unrelated
The selection changes frequently but the basic...	basic	**related**
The selection changes frequently but the basic...	dishes	**related**
The selection changes frequently but the basic...	are	unrelated
The selection changes frequently but the basic...	always	unrelated
The selection changes frequently but the basic...	available	unrelated
The selection changes frequently but the basic...	.	unrelated

Given

```
{'categories': [['food', 'neutral']],
 'targets': [[[4, 13], 'selection', 'neutral'],
  [[41, 53], 'basic dishes', 'neutral']],
 'text': 'The selection changes frequently but the basic dishes are always available.'}
```

Pre-processing

4-class classification
{positive, negative, neutral, conflict}

Aspect Sentiment

Sentence (Sentence 1)	Aspect (Sentence 2)	Label
The selection changes frequently but the basic...	selection	**neutral**
The selection changes frequently but the basic...	basic dishes	**neutral**

ground truth terms

Given


```
{'categories': [['food', 'neutral']],
 'targets': [[[4, 13], 'selection', 'neutral'],
  [[41, 53], 'basic dishes', 'neutral']],
 'text': 'The selection changes frequently but the basic dishes are always
available.'}
```

Pre-processing

Binary classification

Category Extraction

Sentence (Sentence 1)	**Category (Sentence 2)**	**Label**
The selection changes frequently but the basic...	food	**related**
The selection changes frequently but the basic...	service	unrelated
The selection changes frequently but the basic...	price	unrelated
The selection changes frequently but the basic...	ambience	unrelated
The selection changes frequently but the basic...	anecdotes/miscellaneous	unrelated

fixed 5 for each Sentence 1

Given

```
{'categories': [['food', 'neutral']],
 'targets': [[[4, 13], 'selection', 'neutral'],
  [[41, 53], 'basic dishes', 'neutral']],
 'text': 'The selection changes frequently but the basic dishes are always available.'}
```

4-class classification
{positive, negative, neutral, conflict}

Category Sentiment

Sentence (Sentence 1)	Category (Sentence 2)	Label
The selection changes frequently but the basic...	food	**neutral**

ground truth categories

Given

```
{'categories': [['food', 'neutral']],
 'targets': [[[4, 13], 'selection', 'neutral'],
  [[41, 53], 'basic dishes', 'neutral']],
 'text': 'The selection changes frequently but the basic dishes are always available.'}
```

Aspect Extraction
+
Aspect Sentiment

Pre-processing

5-class classification
{none, positive, negative, neutral, conflict}

Sentence (Sentence 1)	**Aspect (Sentence 2)**	**Label**
The selection changes frequently but the basic...	The	none
The selection changes frequently but the basic...	selection	**neutral**
The selection changes frequently but the basic...	changes	none
The selection changes frequently but the basic...	frequently	none
The selection changes frequently but the basic...	but	none
The selection changes frequently but the basic...	the	none
The selection changes frequently but the basic...	basic	**neutral**
The selection changes frequently but the basic...	dishes	**neutral**
The selection changes frequently but the basic...	are	none
...	...	...

Aspect Extraction
->
Aspect Sentiment

Pre-processing

Binary classification

Sentence (Sentence 1)	**Aspect (Sentence 2)**	**Label**
The selection changes frequently but the basic...	The	unrelated
The selection changes frequently but the basic...	selection	**related**
...	...	...

predictions

4-class classification {positive, negative, neutral, conflict}

Sentence (Sentence 1)	**Aspect (Sentence 2)**	**Label**
The selection changes frequently but the basic...	predicted terms	neutral
...	...	...

Given

```
{'categories': [['food', 'neutral']],
 'targets': [[[4, 13], 'selection', 'neutral'],
  [[41, 53], 'basic dishes', 'neutral']],
 'text': 'The selection changes frequently but the basic dishes are always available.'}
```

Pre-processing

Category Extraction
+
Category Sentiment

5-class classification
{none, positive, negative, neutral, conflict}

Sentence (Sentence 1)	Category (Sentence 2)	Label
The selection changes frequently but the basic...	food	**neutral**
The selection changes frequently but the basic...	service	none
The selection changes frequently but the basic...	price	none
The selection changes frequently but the basic...	ambience	none
The selection changes frequently but the basic...	anecdotes/miscellaneous	none

fixed 5 for each
Sentence 1

Category Extraction
->
Category Sentiment

Pre-processing

Binary classification

Sentence (Sentence 1)	**Aspect (Sentence 2)**	**Label**
The selection changes frequently but the basic...	food	unrelated
The selection changes frequently but the basic...	service	**related**
...	...	...

predictions

4-class classification {positive, negative, neutral, conflict}

Sentence (Sentence 1)	**Aspect (Sentence 2)**	**Label**
The selection changes frequently but the basic...	predicted categories	neutral
...	...	...

BERT-models

Models

Models

a
b
a+b
c
d
c+d

Model	Epochs	Batch Size	Learning Rate
Small BERT	10	24	0.00002
BERT	4	16	0.00002
Distil BERT	4	24	0.00002
Large BERT	4	24	0.00002

BERT-models

Performances

Performances: Naive vs BERT for Aspect Extraction and Choose the best BERT

- Task: **Aspect Extraction**
- Dataset: **Restaurants**

Model	Best F1
Naive	67
Distil BERT	**83**

Transformer based architecture is preferable

- Task: **Aspect Extraction**
- Dataset: **Restaurants**

Model	Best F1
Small BERT	79
BERT	81
Distil BERT	**83**
Large BERT	**83**

Distil BERT is preferable

- Dataset: **Restaurants**

Task	Distil BERT	Large BERT
b	55	**60**
a + b	44	**48**
c + d	48	**56**

Large BERT is preferable

Performances: Choose best model for joined and separate tasks, merge datasets

- Dataset: **Restaurants**

Task	->	+
ab	**48**	40
cd	50	**56**

For Aspect Extraction + Aspect Sentiment a -> b performs better than a + b

For Category Extraction + Category Sentiment c + d performs better than c -> d

- Dataset: **Restaurants**

Task	As is	From Combined
a	**83**	82
b	**60**	50
c	**82**	82
d	**64**	55

Separately implemented models perform better than their estimations from a + b and c + d

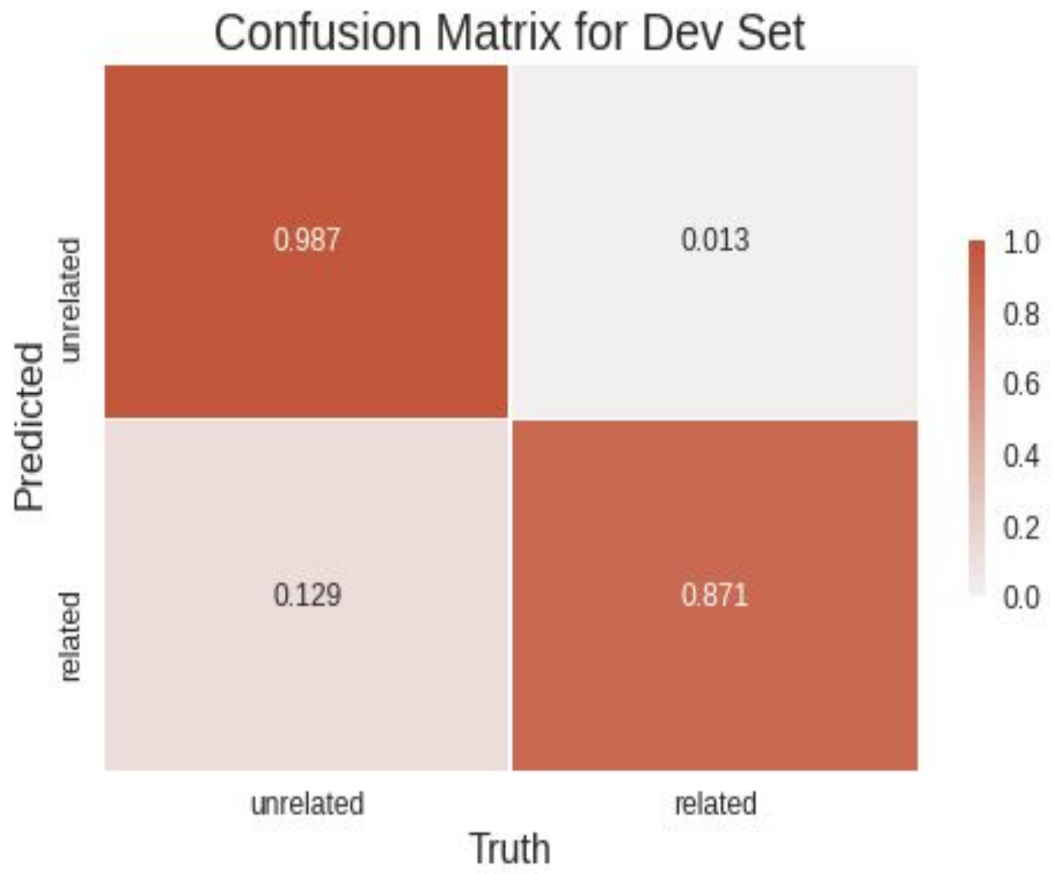
- Task: **Aspect Sentiment**

Dataset	Best F1
Restaurant	60
Laptops	54
Mixed	59

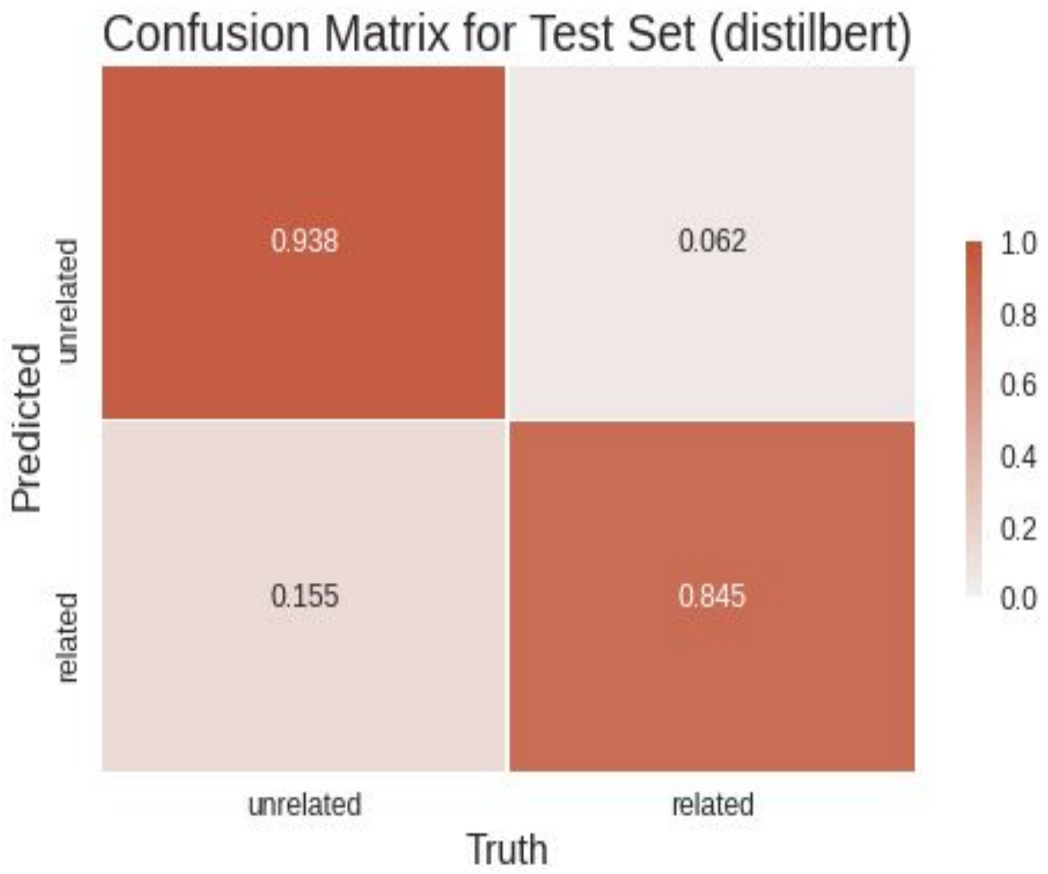
Mixed dataset can be approximated by Restaurants
Laptops are a bit harder than Restaurants

Performances: Confusion matrices for Extractions

For tokens!

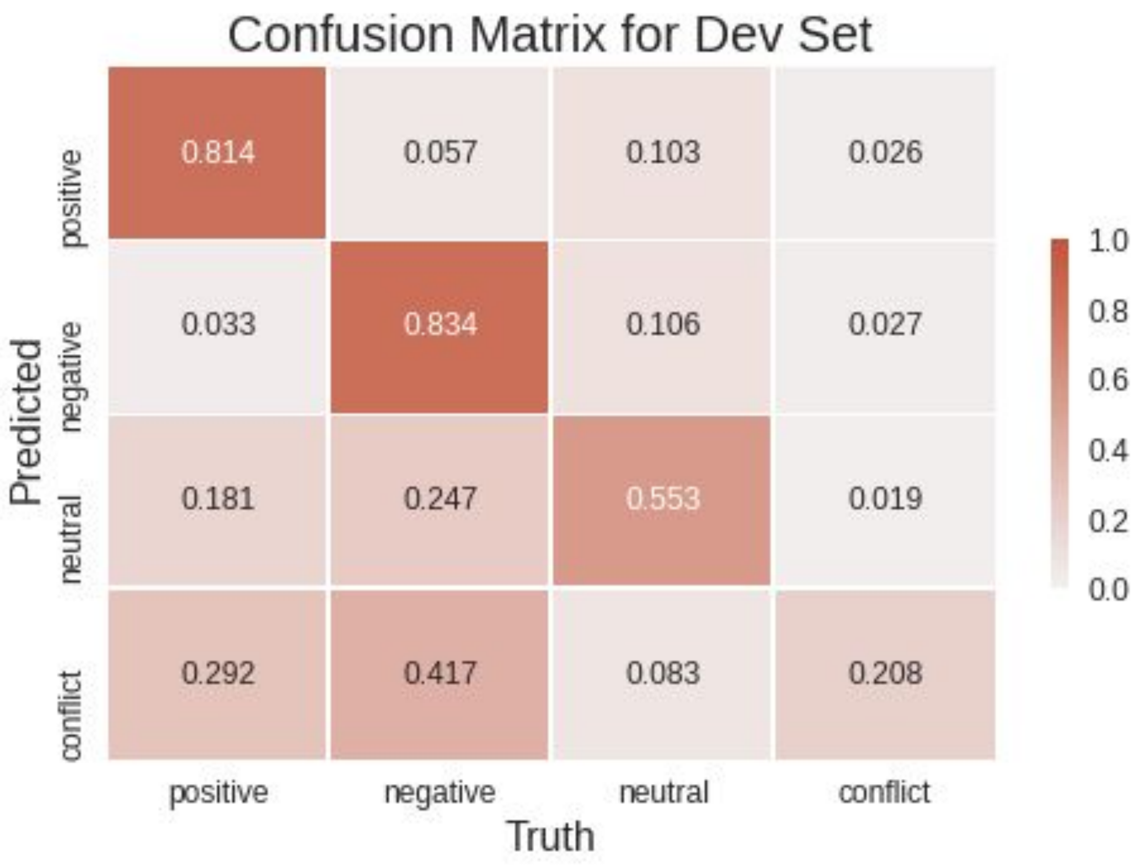


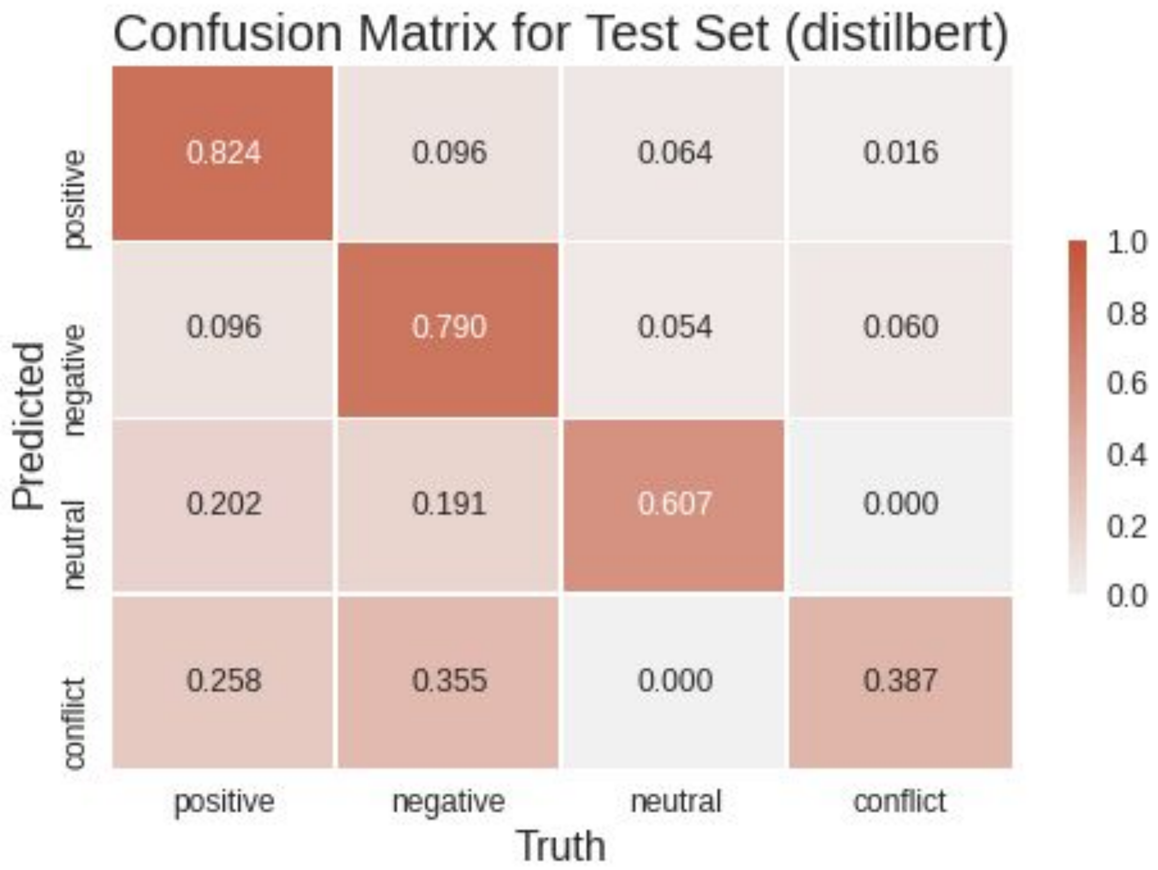
- Task: **Aspect Extraction**
- Dataset: **Mixed**



- Task: **Category Extraction**
- Dataset: **Restaurants**

Performances: Confusion matrices for Sentiments



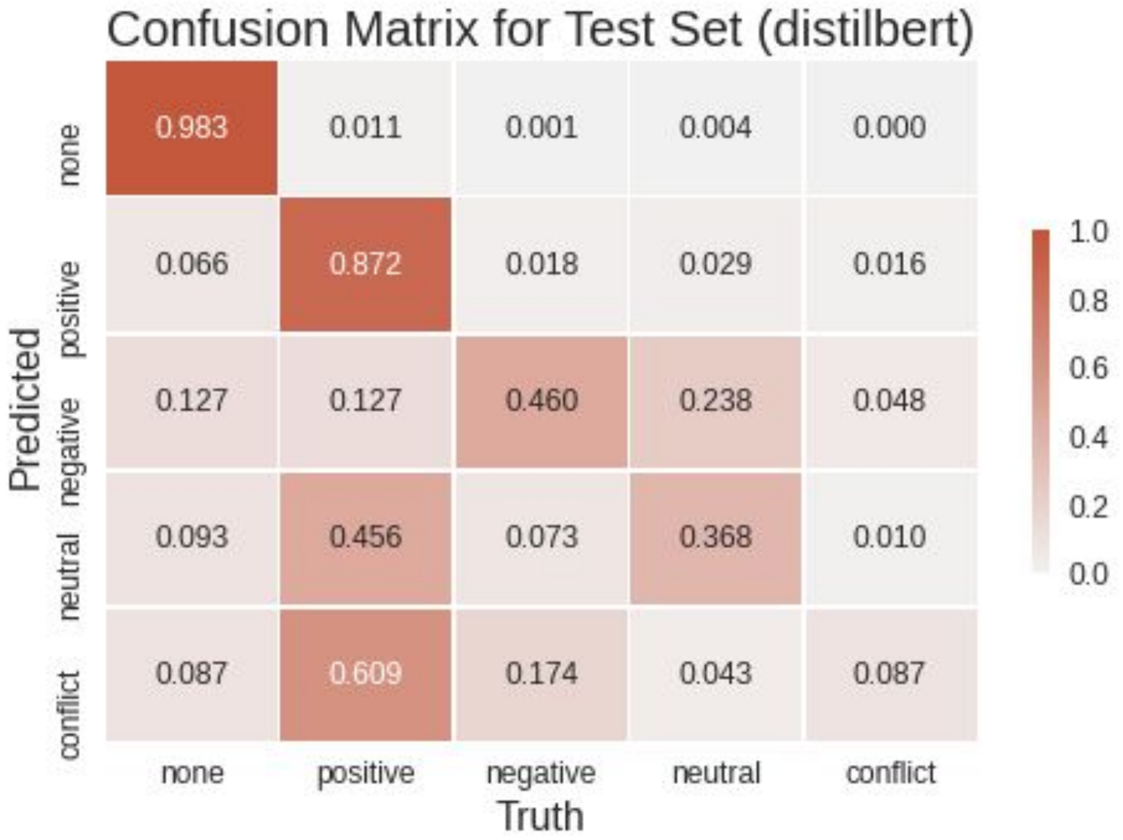


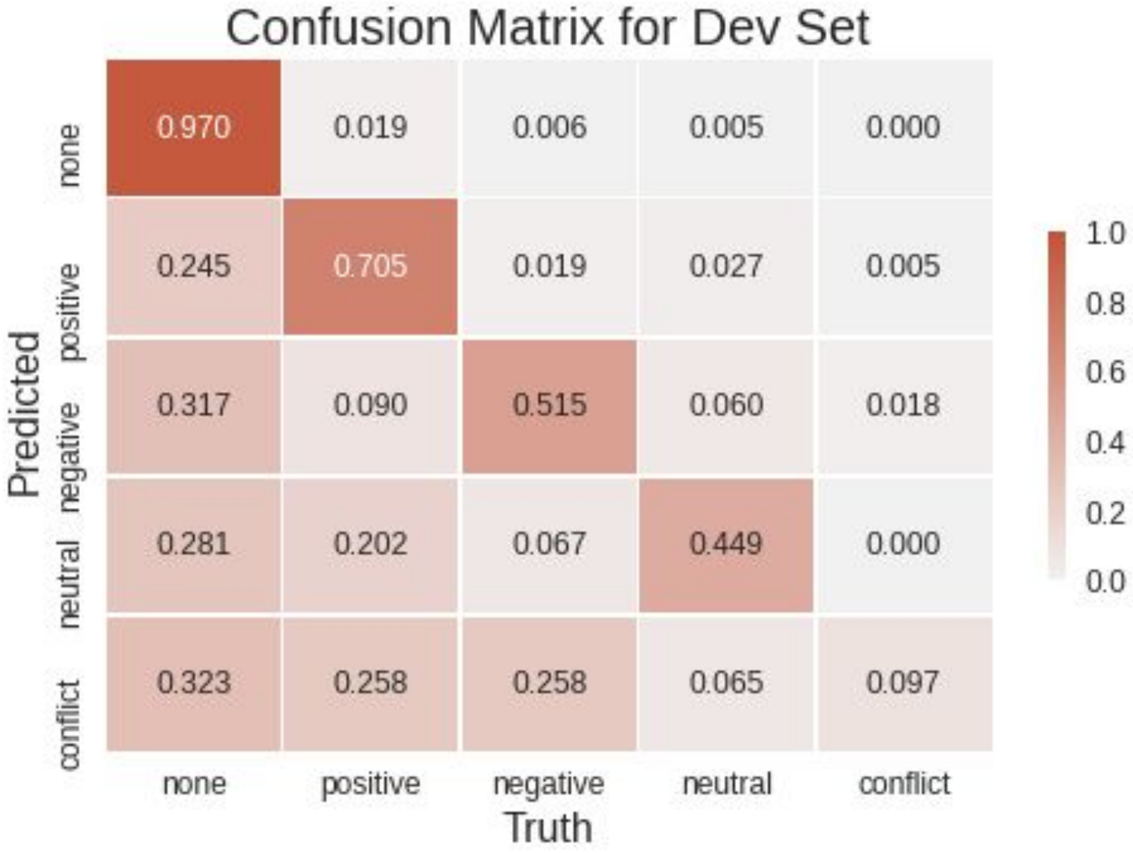
- Task: **Aspect Sentiment**
- Dataset: **Mixed**

- Task: **Category Sentiment**
- Dataset: **Restaurants**

Performances: Confusion matrices for Combinations

For tokens!





- Task: **a + b**
- Dataset: **Mixed**

- Task: **c + d**
- Dataset: **Restaurants**

Conclusion

- **Transformers** are powerful instrument in approaching Aspect-Based Sentiment Analysis;
- **Sentence-pair classification** interpretation of a task is appropriate for use of pre-trained BERT models;
- Work on aspects could be improved through the modification of the **decoding** approach;
- Work on categories could be improved through the **conceptual** modifications;
- Precise **fine-tuning** should be under focus;
- For sentiment analysis part **conflict** class is the hardest for recognition.