TRANSFORMER ARCHITECTURES FOR FINE-GRAINED SENTIMENT ANALYSIS

Advanced topics in Computer science
Project

Michele Bortone

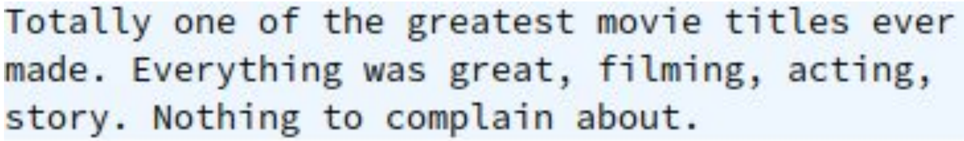
UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
MATEMATICA

Dipartimento di Matematica "Tullio Levi-Civita"

Task

- Sentiment analysis:

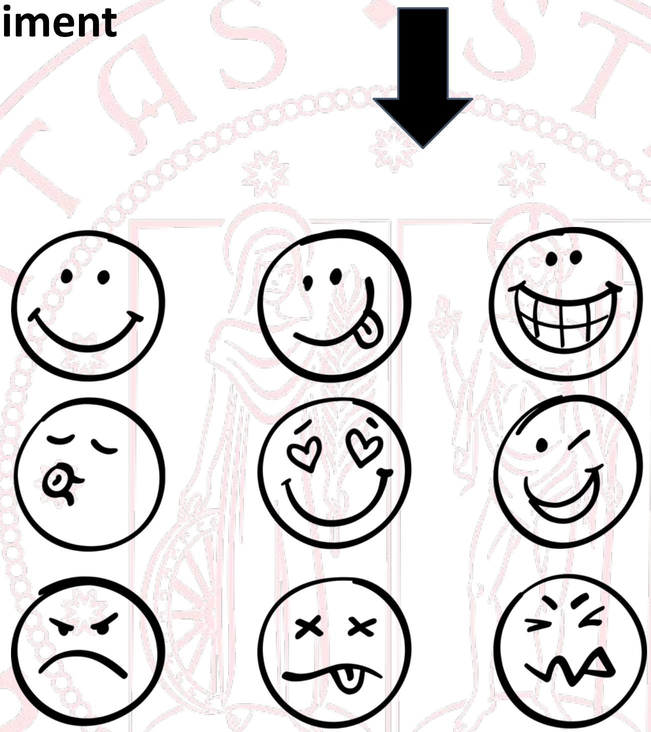


```
Totally one of the greatest movie titles ever
made. Everything was great, filming, acting,
story. Nothing to complain about.
```

  - **Classification** task
  - Given an input text, we want to **predict the sentiment label**

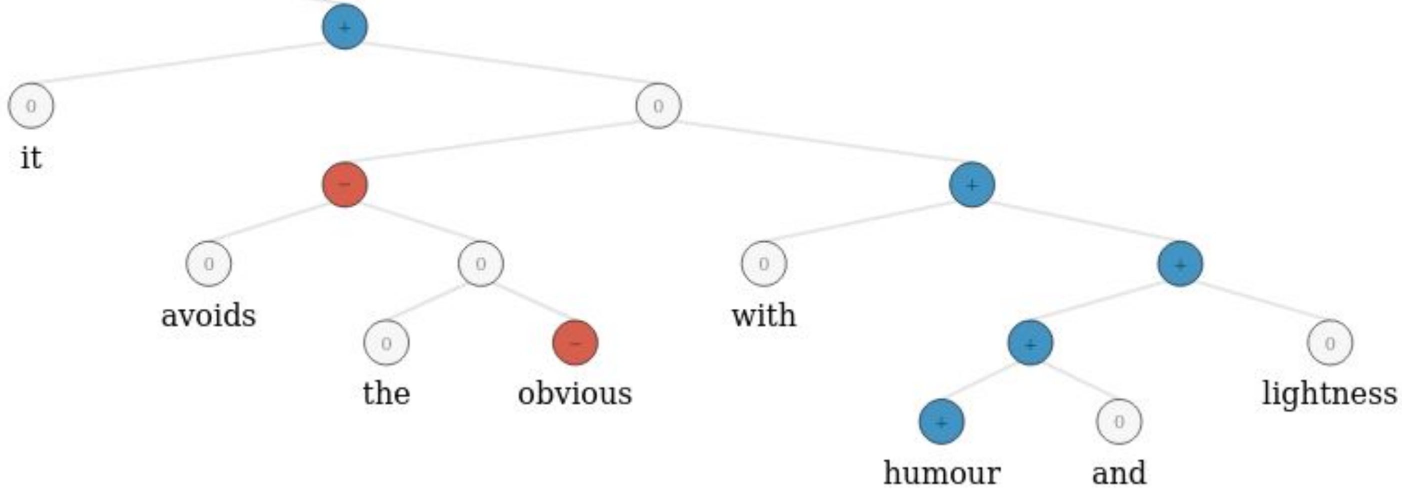
- Two different versions:
  - Binary (Positive, Negative)
  - **Fine-grained** (more than two labels)

FOCUS ON FINE-GRAINED SENTIMENT ANALYSIS but the text is organized with tree structure



Dataset

- Stanford Sentiment Treebank (SST-5)
  - Movie review sentences labelled with **5 classes**
  - Each sentence is represented in a **parse tree**
  - Each node represents a **phrase** and is labelled
  - The **root** node represent the entire sentence
  - We can exploit **context** information by the tree structure



Dataset

- Yelp-5
    - Reviews and number of **stars assigned by the reviewer**
    - **NO tree structure**
    - Reviews can be **very long**

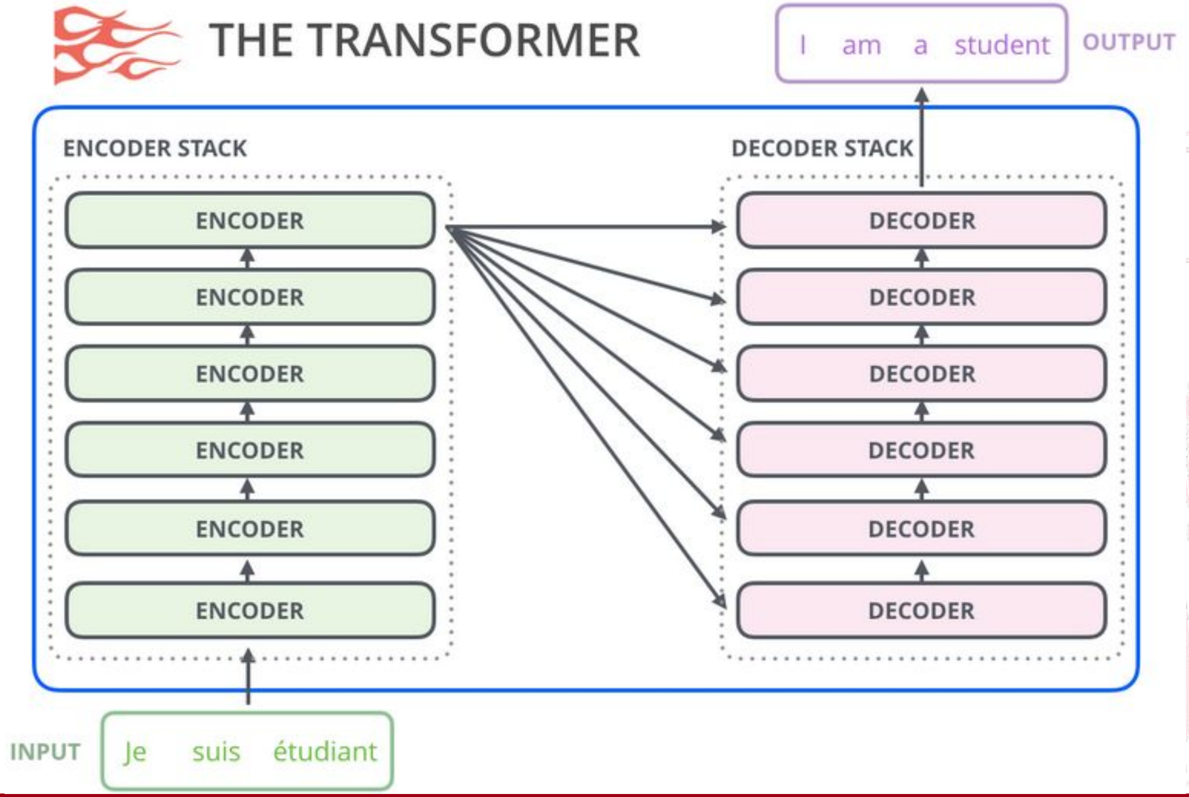
SST-5 is the main dataset. Yelp-5 is a similar dataset in order to do some experiments increasing the size of SST-5

Related work

- **Published works:**
  - **RNTN:** *Recursive deep models for semantic compositionality over a sentiment treebank. 2013.*
  - **BERT:** *Fine-grained sentiment classification using bert, 2019.*
  - **RoBERTa:** *Self-explaining structures improve nlp models, 2020.*
- In the last years **transfer learning** (pretraining and finetuning) and **Transformer** architectures has improved performances.
- **My approach**
  - **BERT:** based on Transformer **Encoders**
  - **GPT2:** based on Transformer **Decoders**

Transformer

- Original architecture for machine translation
- Stack of Encoders and then a stack of Decoders

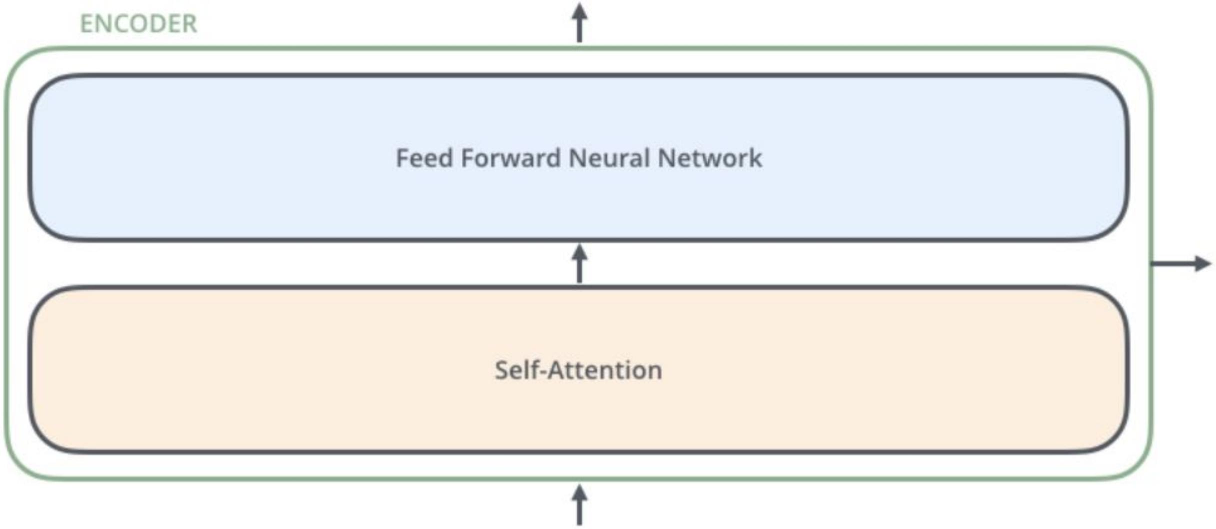


Attention

- Attention mechanism **forces model to focus on specific tokens** in order to capture the context information

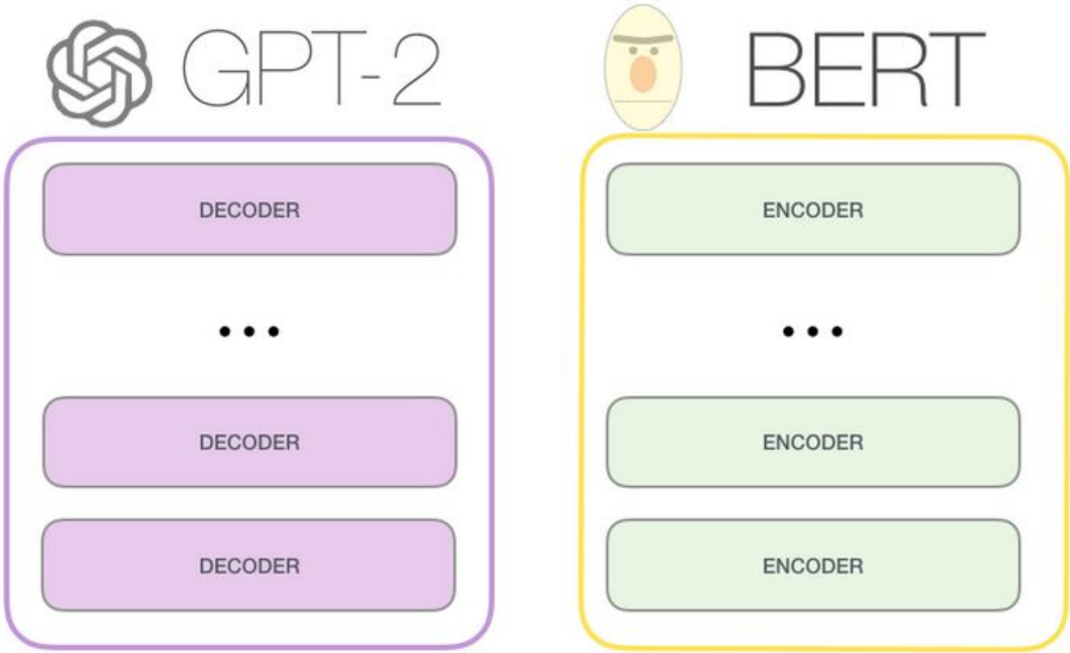
- There are different implementations of Attention for different models

- Attention layers are followed by a **feedforward layer** that produces the output for the stack

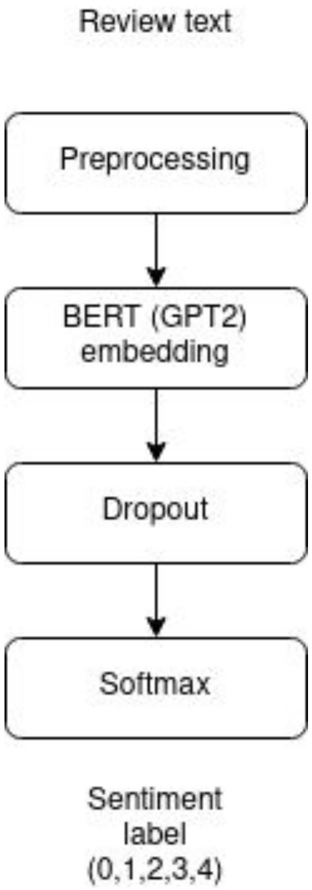


BERT and GPT2

- **Language models** with the same base but **different approaches**
- The main difference is the way in which they perform **Attention**: GPT2 hides tokens at the right of the current step



Classifier



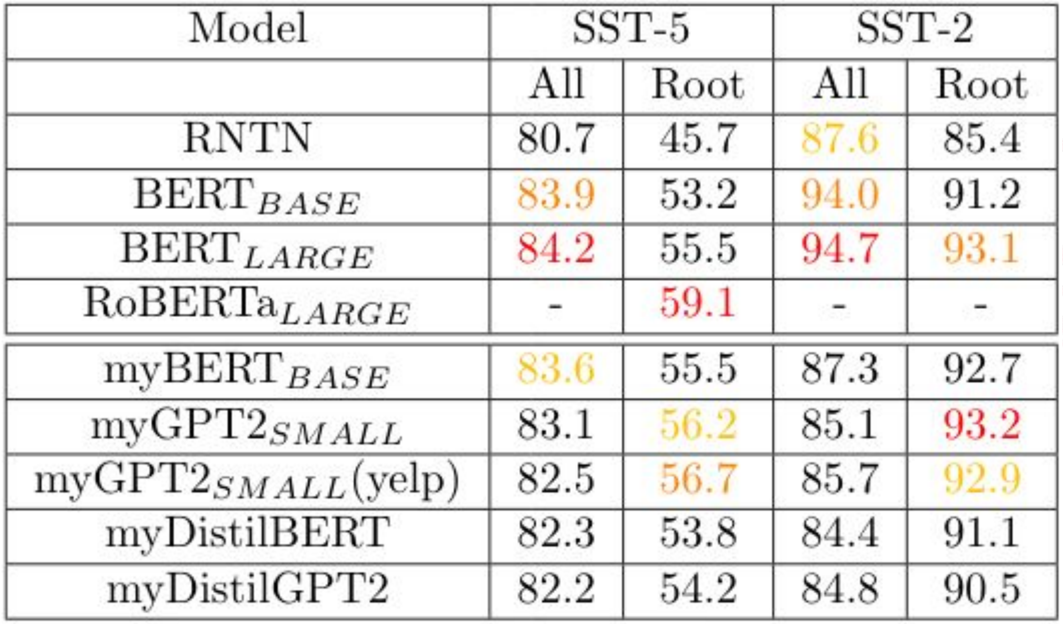
- Text classification:
  - **Preprocessing**: Starting from the tree, compute all subsentences represented by a node
  - **Embedding**: Text tokenization, padding, special tokens addition and embedding with GPT2 or BERT
  - **Dropout**: Addition of dropout layer to avoid overfitting
  - **Classifier**: Softmax to score each class
- The main **difference of data preprocessing** is the addition of padding to the left fort BERT and to the right for GPT2

Experiments

- Target of the experiments:
  - Which model has the **best performance**
  - How performance changes in relation to the **number of parameters**
  - How performance changes if we combine **more datasets**

Results

- Models tested in 4 different tasks:
  - SST-5 all nodes
  - SST-5 root node
  - SST-2 all nodes
  - SST-2 root node
- Models size:
  - LARGE: 340M
  - BASE or SMALL : 110M
  - DISTILLED: 70M



Model	SST-5		SST-2	
	All	Root	All	Root
RNTN	80.7	45.7	87.6	85.4
BERT$_{BASE}$	83.9	53.2	94.0	91.2
BERT$_{LARGE}$	84.2	55.5	94.7	93.1
RoBERTa$_{LARGE}$	-	59.1	-	-
myBERT$_{BASE}$	83.6	55.5	87.3	92.7
myGPT2$_{SMALL}$	83.1	56.2	85.1	93.2
myGPT2$_{SMALL}$(yelp)	82.5	56.7	85.7	92.9
myDistilBERT	82.3	53.8	84.4	91.1
myDistilGPT2	82.2	54.2	84.8	90.5

Red: Best performance
Orange: 2nd performance
Yellow: 3rd performance

Considerations

- **RoBERTa is the state of the art** in SST-5 for the root nodes. Probably it is for the other tasks as well because is the hardest task, but we don't have results
- **GPT2 is the best alternative** with less than 70% of weights
- BERT outperforms GPT2 is we consider all nodes
- Distilled versions are very close to base versions
- Yelp-5 improved performances about 0.5%, so it's not so useful

Conclusion

- All **Transformer architectures** tested **outperformed RNTN**
- Attention mechanism make difference in sentiment analysis
- If we don’t care about the model size, increasing the number of Transformer blocks can increase performance.
- Distilled version offers good **tradeoff between size and performance**