Project Management

Freitag, 23. April 2021 12:49

Description

Topics

- Captioning = Ein Bild mit einer Überschrift versehen
  - Anwendung NLP und Computer-Vision

- Datasets
  - 8.000 Pictures
    - manual selected from six differnet Flickr groups
    - depicting (zeigen) different scenes and situations, NOT well-known people or loacations
  - 5 captions (Beschreibungssätze) per Picture, that describe the salient (hervorstechenden) Entities (Objekte, Personen) and Ev ents (Vorgänge)
  - Download location: https://www.kaggle.com/adityajn105/flickr8k
    - ☐ Folder holding the pictures
    - ☐ Text file containing the captions

- Mission (Arbeitsauftrag), Deliverables (zu liefernte Artefakte) und Todos
  - ☑ Preprocessing
    - ☑ Pictures
      Inception mit pictures ausführen und Output abspeichern (Rainer)
      Results: Dataset + Funktion
    - ☑ Captions (Rainer)
  - ☐ Build an Encoder-Decoder Architecture
    (Encoder: CNN for picture processing, Decoder: RNN for caption processing)
    - ☑ Must: Baseline Model (Andras) ohne Attention
    - ☐ Must: Encoder/Decoder using Attention-Mechanism (Andras)
    - ☐ Nice2Have: Encoder Part: Object recognition via multi-class classification, captions transfromed to entities for recoginition (Thomas)
    - ☐ Nice2Have: Encoder/Decoder using Attention Mechanism, Variation CNN Part (other Network, other dimension of output)
    - ☐ Nice2Have: Encoder/Decoder using Attention Mechanism, Variation RNN Part (Glove-Embeddings, …) (Rainer)
    - ☐ Nice2Have: Encoder/Decoder using Attention Mechanism, Hyperparamter Optimization (learning-rate, layer-units)
    - ☐ Bounty: Build up architecture using Auto-Keras (Rainer), documentation of results backup-page in presentation
    - ☐ ~~Bounty CNN/RNN/Reinforcement Learning~~
  - ☐ Visualize the findings using teached packages
    - ☐ Mathplotlib
    - ☐ Plotly
    - ☐ Keras
    - ☐ Tensorboard (incorporate callback) (Thomas)
    - ☐ Store to disk: Models and history of model run (all)
    - ☐ Talos
  - ☑ Use Metrics (see below)
    - ☑ Evaluation Function für beide Metriken implementieren (Thomas)
  - ☐ Obtain (erhalte) statistics that corroborate (erhärten, bestätigen) the results
    - ☐ Loss per epoch
      - ☐ Training-/Validation-Curves
    - ☐ Learning rate per epoch
    - ☐ Metrics per epoch
      - ☐ Training-/Validation-Curves
  - ☐ Explain the reasons to choosing that statistics
  - ☐ Deliverables
    - ☐ Abgabetermin: 30.04.2021 EOB

- ☐ Abgabe aller Artefakte in einer ZIP-Datei mit dem Namen "AIDA2-DKFI-3_Image-Captioning_Englisch_Baligacs_Schremser.zip"
- ☐ Verzeichnis der ZIP-Datei: https://drive.google.com/drive/folders/1zEQqPRWEcfJ7FEnEht99b-5rckW4FI_Z?usp=sharing

- ☐ PDF Report in english
  - ☐ Thoughts an decisions
  - ☐ Division of work among the team members (Arbeitsaufteilung)

- ☐ Jupyter Notebooks and related ressources to reproduce the work -> Cookie-Cutter project, without data

- ☐ Presentation in english 10 - 15 minutes
  - ☐ Process description
  - ☐ Approaches used
  - ☐ Reasons for choosing the approaches
  - ☐ Results and findings

Short Sentences

- Reference: “Back to the Future” premiered 30 years ago
- MT: “Back to the Future”

- 1-gram: 4/4
- 2-gram: 3/3
- 3-gram: 2/2
- 4-gram: 1/1

$$BP = \begin{cases} 1 & ,c > r \\ e^{1-r/c} & ,c \le r \end{cases}$$

$$BLEU = \sqrt[4]{P_1 * P_2 * P_3 * P_4} * BP$$

- Brevity Penalty: $e^{1-\frac{8}{4}} = e^{-1} = 0.37$

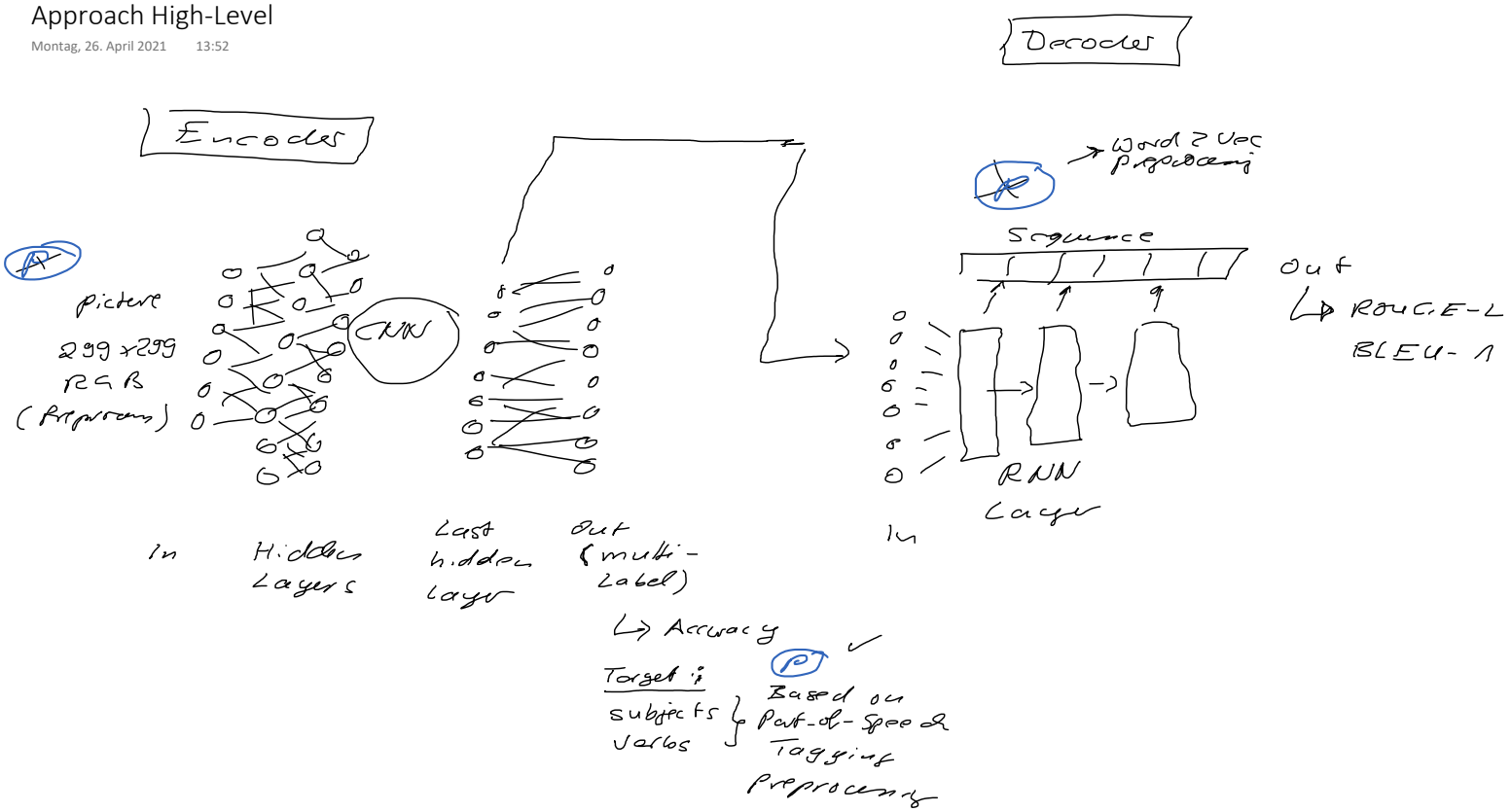
- Reference 1: the new movie
- Reference 2: the new film
- MT: the the

- P_1= ~~2/2~~ 1/2

- Clipping:
  - Max count of of n-gram in any reference

Approach High-Level

Montag, 26. April 2021 13:52



BaselLine

Dienstag, 27. April 2021 10:00

CNN: InceptionV3
RNN: LSTM without Attention

Metriken

Dienstag, 27. April 2021 11:11

Metrics

Allgemein: https://stackoverflow.com/questions/38045290/text-summarization-evaluation-bleu-vs-rouge

- ROUGE-L (Recall-Oriented Understudy of Gisting Evaluation)

https://ilmoirfan.com/rouge-an-evaluation-metric-for-text-summarization/

Longest Common Subsequence (LCS) based statistics.
Longest common subsequence problem takes into account sentence level structure similarity naturally and identifies longest co-occurring in seqence n-grams automatically

ROUGE-L – measures longest matching sequence of words using LCS. An advantage of using LCS is that it does **not require consecutive matches but in-sequence matches** that reflect sentence level wordorder. Since it automatically includes longest in-sequence common n-grams, you don't need apredefined n-gram length.

Vergleich von 2 Sätzen: X = Referenzssatz, Y = machinell erzeugter und zu vergleichender Satz
ROUGE-L = 0 -> X und Y haben nichts gemeinsam
ROUGE-L = 1 -> X = Y, die beiden verglichenen Sätze sind gleich

Vorteile:

- Es werden nur Sequenzen von Wörtern beurteilt und nicht Wörter einzeln
- Es sind keine vordefinierte Sequenzen von sog. n-gramms notwendig, da der Algorithmus die längsten Wortsequenzen selbst findet

Precision, Recall and F-measure

To evaluate how accurate our machine generated summaries are we compute the Precision, Recall and F-measure for any of this metric.

In ROUGE **recall** refers that how much words of candidate summary are extracted from reference summary. Formula to calculate recall:

$$R = \frac{Nnumber\ of\ overlaping\ words}{Total\ words\ in\ reference\ summary}$$

For example, recall for unigram in the below example:

R1- The dog bites the man.

S1- The man was bitten by the dog, find in dark.

$$\frac{4}{5} = 0.8$$

It shows that almost all words in candidate (machine generated) summary have been extracted from reference summary. It means our system generated a good summary that is exactly same as reference summary.

But it is not always a good case sometimes the machine generated summary may be too long and contains most of the irrelevant words. So, it may not be a good summary. If the size of machine generated summary is predefined then recall alone may provide the enough information (candidate summary is relevant or not).To find the other case (if machine generated summary is good or not) we also need to compute precision.

In ROUGE **precision** refers that how much candidate summary words are relevant. Formula to calculate recall:

$$P = \frac{Nnumber\ of\ overlaping\ words}{Total\ words\ in\ candidate\ summary}$$

In above example:

$$\frac{4}{10} = 0.4$$

F measure provides the complete information that recall and precision provides separately.

$$F - measure = \frac{(1 + \beta^2)R*P}{R + \beta^2*P}$$

$\beta = 1$ so.

$$F_1 = 2.0 * \frac{0.8*0.4}{0.8 + 0.4} = 0.53333$$

Beispiel:
X = police killed the gunman
Y1 = police kill the gunman -> ROUGE-L (bei β=1) = 3/4 = 0,75 -> longest sequence: police the gunman

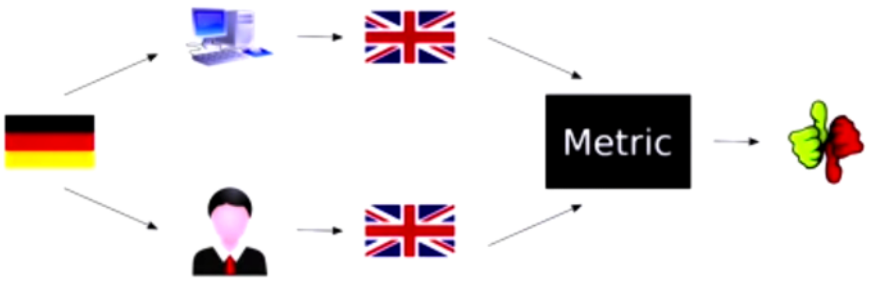
Y2 = the gunman kill police -> ROUGE-L (bei β=1) = 2/4 = 0,5 -> longest sequence: the gunman

In Y1, the first word and last two words match the reference, so it scores 3/4, whereas Y2 only matches the bi-gram, so scores 2/4.
Aus <https://stats.stackexchange.com/questions/301626/interpreting-rouge-scores>

- BLEU-1 (bilingual evaluation understudy)

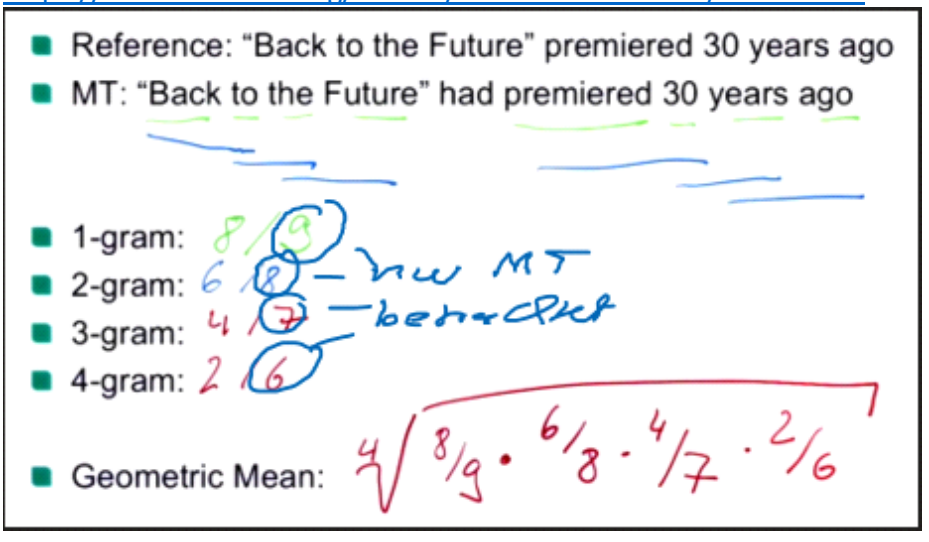
BLEU-1 sagt aus, dass der Score für Unigramms (1-gram) betrachtet wird?

Es wird die maschinelle Übersetzung mit menschlichen Übersetzungen verglichen. Hierbei werden allgemeingültige Satzsegmente im maschinelle Text mit qualitative guten Referenzsätzen (mehrere!) der menschlichen Übersetzung verglichen.



BLEU = 1, sehr gute Übereinstimmung
BLEU = 0, keine Übereinstimmung

Beispiele:
https://www.coursera.org/lecture/machinetranslation/bleu-Bv81F



- Reference 1: students said they looked forward to his class
- MT: students said they were excited about his lecture
- $P_1 = 0/8$

- Reference 1: students said they looked forward to his class
- MT: students said they were excited about his lecture
- P_4= 0/8
- Document level scores:
  - Aggregate statistics over whole document

BLEU

- Matches exact words
  - Several references possible
- Adequacy: Modeled by word precision
- Fluency: Modeled by n-gram precisions
- No recall: only Precision is calculated
  - „brevity penalty“ to prevent short sentences
- Calculate aggregate score over a large test set

***n*-gram** is a contiguous sequence of *n* items from a given sample of text or speech. The items can be phonemes, syllables, letters, words or base pairs according to the application

Aus <https://en.wikipedia.org/wiki/N-gram>

Implementierung ROUGE-L:
Python:
https://pypi.org/project/rouge-score/
https://pypi.org/project/easy-rouge/
https://pypi.org/project/rouge-metric/
https://pypi.org/project/py-rouge/
https://pypi.org/project/rouge/

Implementierung BLEU:
Python
https://pypi.org/project/bleu/
NLTK:
https://machinelearningmastery.com/calculate-bleu-score-for-text-python/
https://stackoverflow.com/questions/32395880/calculate-bleu-score-in-python

Keras:
Es muss eine sogn. Custom-Metric angelegt werden:
https://keras.io/api/metrics/

Implementierung der von uns benötigten Metriken für Keras:
https://github.com/danieljl/keras-image-captioning
Anwendung:

```
import src.models.metrices as met
…
model.compile(optimizer=Adam(lr=self._learning_rate, clipnorm=5.0),
              loss=categorical_crossentropy_from_logits,
              metrics=[met.categorical_accuracy_with_variable_timestep])
…
```

Optimizations

Mittwoch, 28. April 2021 09:43

Encoder

Mittwoch, 28. April 2021 10:50

1. Add Attention
2. enc_output.output -> increase the units from 512 to higher value
3. CNN with object recognition
4. Try another pre-trained net

Decoder

Mittwoch, 28. April 2021 10:51

1. Use Glove-Embeddings
2. Preprocess Captions
    - Texte in lower case transformieren
    - Wörter mit nur einem Buchstaben entfernen (bspw. "a")