Natural Language Generation using structured input
Generating descriptions from textual attributes

Kim Almasan, Tatjana Chernenko, Siting Liang, Bente Nittka

Institute for Computational Linguistics,
Ruprecht-Karls-Universität Heidelberg
{chernenko/liang/nittka}@cl.uni-heidelberg.de

Software Project
Prof. Dr. Anette Frank

Outline

- Introduction
- Approach
  - Model
  - Software Architecture
  - Evaluation Techniques
- Datasets, Experimental Setup and Results
- Best model
- Demo
- Lessons learned, Conclusion and discussion
- Additional Experiments
  - Lessons
  - Questions

Introduction

Automatic image description generation is a problem which involves vision recognition in order to detect objects, substances, and locations.

For our approach we assume that we already have such attributes provided by classifier predictions over the image or any external knowledge in an application

- **Goal:** explore impact of features provided in terms of descriptive attributes to create a model that ultimately generates a textual description that verbalises the detected aspects of the image.

Model Development I

Bidirectional RNN

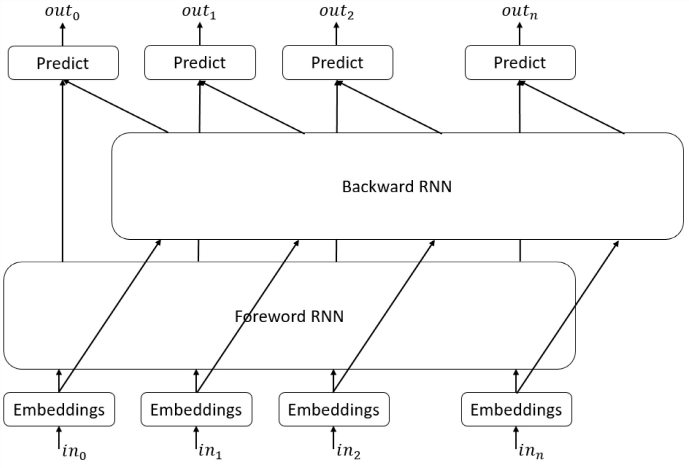


Abbildung: Bi-RNN

Model Development II

Encoder-Decoder

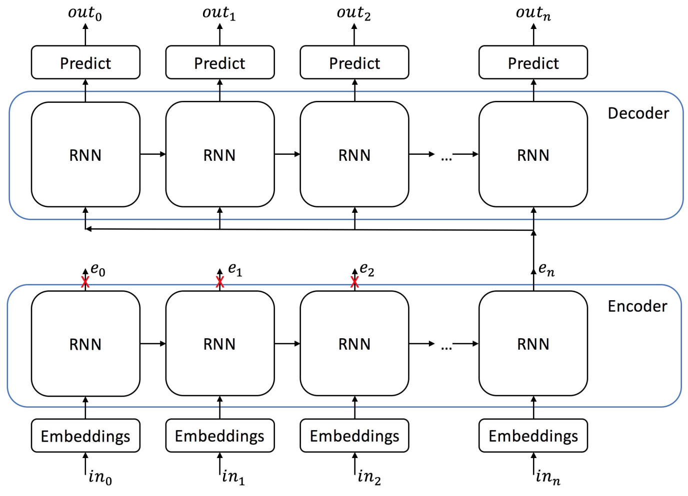


Abbildung: Encoder-Decoder

Model Development III

Encoder-Decoder with Attention

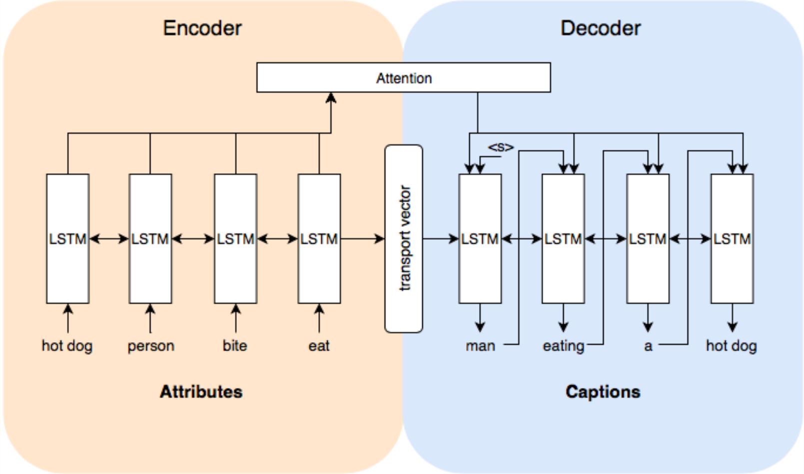


Abbildung: Model

Software Architecture

- **Python 3.6**
  - Typing
  - Future Proof
- **Pipenv**
  - Easy to setup
  - Dependency Management
- **Tensorflow**
  - Industry Standard
  - Good Documentation
- **GIT**
  - Easy Collaboration

Evaluation I

Automatic Evaluation: Metrics

- **Bleu, CIDEr, ROUGE, METEOR, SPICE**
- word-based evaluation, measure word-overlap
- we use implementation provided by MS COCO
- results are only of limited meaningfulness for our task

Flesch reading ease score

- indicates how easy or difficult to read a text is
- we found out that it doesn't provide useful information for the evaluation of our generated captions and thus left it out from the evaluation results

Evaluation II

Human Evaluation

Evaluators (= we) rate caption on a 6-point Likert Scale for the following criteria:

- **Naturalness:** Could the utterance have been produced by a native speaker?
- **Informativeness:** Does the utterance provide all the useful information from the image?
- **Quality:**How do you judge the overall quality of the utterance in terms of its grammatical correctness and fluency?

We followed Novikova et al. when choosing the criteria for evaluation

- human evaluation played a major part in the process of deciding which model performed best
- for the output of every model version, 3 persons evaluated 30 generated captions

Experiment Set 1

Question 1: Input Type

How do the captions improve if we enhance the information in the input?

Three different data sets:

- **MS COCO:** objects: values and categories[LMBBGHPRDZ14]
- **COCO-a:** objects, actions and adverbs: values and categories[RP15]
- **COCO-a:** objects, actions and adverbs: **values only**

COCO-a Attribute Examples

MS COCO image n.248194

MS COCO captions

A man reading a paper and two people talking to a officer.

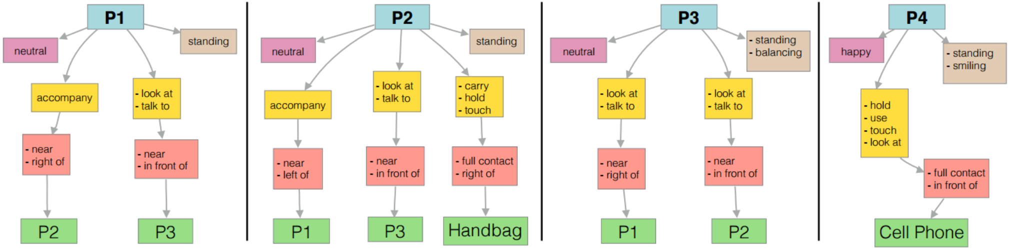
A man in a yellow jacket is looking at his phone with three others are in the background.

A police officer talking to people on a street.

A city street where a police officer and several people are standing.

A police officer who is riding a two wheeled motorized device.

COCO-a annotations (this paper)



System Input Attribute Examples

Example image from the test data

MS COCO

person person person person cake food dining table furniture

COCO-a: values and categories

dining table furniture person person objects use perception look perception touch posture lean location in_front distance light_contact

person person person person contact hold contact hug perception touch social accompany social be_with social dine location left distance light_contact

person person posture sit solo pose solo smile emotion happiness

dining table furniture person person contact pet contact reach objects use perception look perception touch location below location in_front distance full_contact distance light_contact distance near

cake food person person nutrition prepare objects light objects show perception look perception sniff location in_front distance near

Note on COCO-a input length

- Example inputs from COCO-a have been shortened to fit on the slides. Not all interactions are included!
- if technically possible, the input for the COCO-a models in experimental set 1 includes all interactions for each picture
- the maximum input length in the data set is 6218 (519 interactions), the average input length is 224 (20 average interactions)
- due to limited memory the models have been trained with a maximum input length of 500 (cutting off everything after the first 500 tokens)

Question 2: Architecture Choices

How do the captions improve if we support the model with attention and pre-trained embeddings?

Three different versions of the model:

- neither attention nor pre-trained embeddings
- using attention but no pre-trained embeddings
- using attention and embeddings trained with glove

Experiment Set 1. Models Overview

Tabelle: Experiment Set 1

1 SET - low quality Datasets, all interactions included								
Dataset	**BASELINE:** MS_COCO (1.1_2_3)			COCO-a categories + values (1.4_5_6)			COCO-a values only (1.7_8)	
Model #	1	2	3	4	5	6	7	8
Attention	no	yes	yes	no	yes	yes	yes	yes
Embeddings	own	own	GloVe	own	own	GloVe	own	GloVe

Experiment Set 1. Automatic Evaluation

Tabelle: Experiment Set 1, results of automatic evaluation

1 SET - law quality Datasets, all interactions included								
Dataset	**BASELINE:** MS_COCO (1.1_2_3)			COCO-a with categories (1.4_5_6)			COCO-a values (1.7_8)	
Model #	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8
Bleu_1	0.489	**0.628**	0.618	0.492	0.516	0.573	0.616	0.613

- SPICE, CIDEr, Bleu_1,Blue_2, Bleu_3, Bleu_4, ROUGE_L, METEOR, SPICE show the same relative performance, that's why we show **only Bleu_1** scores here
- Automatic Evaluation doesn't say much about the results - use human evaluation
- COCO-a Datasets (1.4_5_6 and 1.7_8) contain **too much input information** (20-500 interactions for one image) at this point: try to tune hyperparameters, reduce interactions number

Experiment Set 1. Manual Evaluation

Tabelle: Experiment Set 1, results of manual evaluation

1 SET - low quality Dataset, all interactions included								
Dataset	**BASELINE:** MS_COCO (1.1_2_3)			COCO-a with categories (1.4_5_6)			COCO-a values (1.7_8)	
Model #	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8
Naturalness	1.129	5.258	5.204	1.495	5.148	**5.086**	5.075	5.0
Quality	1.108	5.591	5.527	1.538	5.574	**5.548**	5.333	5.538
Informativeness	1.581	4.151	4.151	1.817	3.194	**4.28**	3.989	4.086

Conclusion - Experiment Set 1

What model do we improve in the next experiments?

We choose **model 1.6** for the future experiments, because it shows the highest average informativeness scores together with high naturalness and quality scores

Generated Descriptions - Experiment Set 1

Image # 90311

Easy example, big number of similar images in train data. Generated Descriptions:

Model 1.1 - a man player a a a a a

Model 1.2 - a tennis player swinging a tennis racket on a tennis court

Model 1.3 - a man holding a tennis racket on a tennis court

Model 1.4 - a man player a a baseball a a a

Model 1.5 - a woman on a tennis court holding a racket

Model 1.6 - a man is playing tennis on the court

Model 1.7 - a man is hitting a tennis ball with a racket

Model 1.8 - a man in a white shirt is playing tennis

Generated Descriptions - Experiment Set 1

Image # 188657

Difficult example, small number of similar images in train data. Generated Descriptions:

Model 1.1 - a man girl a a of on

Model 1.2 - two men sitting next to each other in a kitchen

Model 1.3 - a man and a woman are sitting on a table

Model 1.4 - two people of people a a a a

Model 1.5 - a group of people sitting around a table topped with wine glasses

Model 1.6 - a woman is cutting a cake as a woman is cutting a cake

Model 1.7 - a woman is eating a pizza with a knife

Model 1.8 - a couple of women sitting in front of a frosted cake

Experiment Set 2

Question 3: Model Tuning

What are the best hyperparameters for our model?

- tune dropout
- tune the number of encoder/decoder layers
- tune beam size
- etc.

Results Experiment Set 2

Tabelle: Experiment set 2

SET # 2 - TUNE HYPERPARAMETERS				
Dataset	1.6 - Coco-a categories and values			
Attention	yes			
Embeddings	Glove			
Model #	**2.1a | 2.1b**	**2.2a | 2.2b**	**2.3a | 2.3b**	**2.4a | 2.4b**
drop out (0.2 *)	0.1 | 0.4			
encoder layers (8 *)		4 | 10		
decoder layers (8 *)			4 | 10	
beam search (0 *)				2 | 5

* - *default values*

Results Experiment Set 2

Tabelle: Experiment set 2

SET # 2 - TUNE HYPERPARAMETERS - BLEU_1				
Dataset	1.6 - Coco-a categories and values			
Attention	yes			
Embeddings	Glove			
Model #	**2.1a | 2.1b**	**2.2a | 2.2b**	**2.3a | 2.3b**	**2.4a | 2.4b**
drop out	0.576 | 0.633			
encoder layers		0.658 | 0.596		
decoder layers			0.512 | 0.645	
beam search				0.608 | 0.624

* - *default values*

Results Experiment Set 2

Tabelle: Experiment set # 2 - Human Evaluation

SET 2 - TUNE HYPERPARAMETERS		
Dataset	1.6 - Coco-a categories and values	
Attention	yes	
Embeddings	Glove	
Model #	**2.1a | 2.1b**	**2.2a | 2.2b**
drop out		
- Naturalness	4.58 | 5.20	
- Quality	5.31 | 5.46	
- Informativeness	4.27 | 4.84	
encoder layers		
- Naturalness		5.11 | 5.13
- Quality		5.29 | 5.55
- Informativeness		3.85 | 4.19

Results Experiment Set 2

Tabelle: Experiment set # 2 - Human Evaluation

SET 2 - TUNE HYPERPARAMETERS		
Dataset	1.6 - Coco-a categories and values	
Attention	yes	
Embeddings	Glove	
Model #	**2.3a | 2.3b**	**2.4a | 2.4b**
decoder layers		
- Naturalness	5.32 | 5.13	
- Quality	5.62 | 5.73	
- Informativeness	3.48 | 4.11	
beam search		
- Naturalness		5.47 | 5.30
- Quality		5.86 | 5.81
- Informativeness		4.11 | 4.66

Conclusion Experiment Set 2 (Hyperparameters)

What hyperparameters should we use in future models?

According to the Experiment Set 2, the best results we get by model 2.1b (droput = 0.4, other hyperparameters = default). We use it in our future experiments. Later this set of hyperparameters is called *improved hyperparameters*.

Experiment Set 3

Question 4: Limited descriptors

How do the captions improve if we limit the number of interactions per image?

- images contain a lot of interactions (up to 513 for one image), only a small fraction of which appear in the captions
- we limit the number of interactions per image by cutting off all except the first few interactions
- problem: we do not know which interactions are relevant for the caption

Results Experiment Set 3

Tabelle: Experiment set 3

3 SET	**3 SET - low quality, max # of interactions=2, 3, 4**		
Dataset	1.6 - Coco-a categories and values		
Attention	yes		
Embeddings	GloVe		
Hyperparameters	improved	improved	improved
Max # of interactions	2	3	4
Model #	3.1	3.2a	3.3
Bleu_1	0.586	0.586	0.591
Naturalness	4.705	4.871	4.705
Quality	5.443	5.451	5.197
Informativeness	3.508	4.032	3.475

Conclusion Experiment Set 3

What is the optimal number of the interactions for input?

The best results within the Experiment Set 3 we get by max 3 interactions in input. However, models from the Experiment Set 1 and Experiment Set 2 show much better results. Furthermore, Experiment Set 3 doesn't show any direct dependency between the number of input interactions and performance, possibly because we don't know which interactions are relevant for the caption.

We don't limit the number of input interactions in future experiments and use model 2.1b for the future experiments.

Experiment Set 4

Question 6: Data quality

How do the captions improve if we increase the quality of the data?

- Amazon Mechanical Turk provides different levels of annotator agreement for their annotations
- so far we used only used data of the lowest quality with attributes produced by one annotator
- for this experiment we want to use only images with attributes where at least three annotators had agreed on the annotations

Results Experiment Set 4

Tabelle: Add caption

4 SET	**4 SET - Dataset of good quality, but less training data**	
Dataset	4.1b (small 1.6 - Coco-a categories and values with 3 Annotators only)	
Attention	yes	
Embeddings	GloVe	
	default	improved
Model #	**4.1**	**4.2**
Bleu_1	0.571	0.652
Naturalness	5.5	5.011
Quality	5.839	5.463
Informativeness	4.645	3.957

Conclusion Experiment Set 4

Do the captions improve if we increase the quality of data?

Data with high level of annotator agreement allows us to win on Naturalness, Quality and on Informativeness of the generated captions. Models with default and improved hyperparameters show a significant improvement of the generated captions, even though the volume of the input training data is smaller.

The model with default hyperparameters shows better results than the model with improved set of hyperparameters, which implies that hyperparameters tuning should be ideally done for each experiment separately (see Additional Experiments section for more details).

High level of annotator agreement is important.

Experiment Set 5

Question 5: More training data

How do the captions improve if we increase the number of training data?

- bigger training set might produce better models
- idea: data augmentation by using back-translation
- translation of the captions from English to German and back
- attributes stay the same
- we want to train the model 2.1b on the augmented data (low quality Coco-a Dataset with captions and values, augmented with Back Translation)

Results Experiment Set 5

Tabelle: Add caption

5 SET	**5 SET - Augmented Dataset 1.6 (Coco-a values and categories)**
Dataset	Augmented 1.6 - Coco-a categories and values
Attention	yes
Embeddings	GloVe
Hyperparameters	improved
Max # of interactions	default
Model #	**5.1**
Bleu_1	0.665
Naturalness	4.516
Quality	4.925
Informativeness	4.075

Conclusion Experiment Set 5

Do the captions improve if we have more training data?

Experiment set 5 shows that more training data improves Blue 1 score, but leads to degradation of Naturalness, Quality and Informativeness of the captions.

High level of annotator agreement is more important, than more input training data.

Best Model

What settings show the best results?

Model 4.2 shows the best results:
Bleu_1 = 0.571, Naturalness = 5.5/6, Quality = 5.839/6, Informativeness = 4.645/6.

Settings:

- Dataset: Coco-a Dataset with categories and values
- Number of interactions: unlimited (but max 500 interactions per image)
- Hyperparameters: dropout = 0.4, encoder layers = 8, decoder layers = 8, beam search = 0, learning rate = dynamic
- Attention: yes
- Embeddings: GloVe
- Data Quality: high Quality
- More training Data: No (not augmented)

Best Model - Discussion

What settings show the best results?

- Categories of the detected objects/actions/events/etc. of the image in the input data help to improve the performance
- Limited number of input interactions doesn't help, if we don't know, what interactions are relevant for the image
- Attention helps to improve the results significantly
- Pre-trained word embeddings (GloVe) work much better as own embeddings (because of the small volume of input data)
- Data quality (high annotator agreement score) is more important, than more training data

Additional Experiments

What can we do with hyperparameters?

- As the time of the project is limited, we cannot make the proper tuning of all the hyperparameters and their combinations
- The combination of the hyperparameters working good for the one model and dataset doesn't always mean that this combination works good for the next models
- It could be helpful to tune the hyperparametrs for the every chosen model separately
- To demonstrate the effect of hypermarameters we perform an additional experiment:
  - Chose two best hyperparameters from Experiments Set 2 *
  - Test the combination of these hyperparameters on the settings of the Experiment Set 2 (unlimited number of interactions in input)
  - Do the same on the settings of the Experiment Set 3 (with max 3 interactions in input)

**dropout = 0.4, encoder layers = 10, other hyperparameters default*

Additional Experiments

Tabelle: Additional Experiment Set

EXTRA	**Influence of hyperparameters**	
Dataset	1.6 - Coco-a categories and values	
Attention	yes	
Embeddings	GloVe	
Hyperparameters	2.1a + 2.1b	
Max # of interactions	unlimited	3
Model #	**2.1b+2.2.b**	**3.2b**
Bleu_1	0.57	0.569
Naturalness	4.57	5.602
Quality	5.30	5.817
Informativeness	3.98	4.462

Additional Experiments - Discussion

When and how should we tune hyperparameters?

The Additional Experiment Set shows that example combination of the best hyperparameters doesn't work on the tested Dataset, performing much worse, as these hyperparameters separately. Otherwise, this combination shows high results on the Dataset with limited number of interactions.

We suppose that our best model could reach even better results if a proper tuning of the hyperparameters will be done not only at the beginning of the experiments, but also at the end.

Demo Time

DEMO TIME

Discussion I

What can our model do?

- model is able to produce grammatical captions
- still has some problems with learning semantically correct captions
- captions are in parts a very good description of the image and partly completely unrelated nonsense, depending on the difficulty
- our best model (using high quality input data) is able to solve the above mentioned problems and to produce grammatical, semantically correct captions in most cases

What might it be useful for?

- results show that it is possible to extract useful information from textual attributes when generating captions
- attributes alone don't provide enough information to generate accurate captions in all cases
- but can be useful in combination with other types of input $\rightarrow$ multimodality

Discussion II

Difficulties and possible improvements

- sequential encoder architecture wasn't optimal, input too long
  $\rightarrow$ maybe hierarchical would have worked better
- so far, our model did not really profit from the richer input Coco-a provides
  - the big amount of information from COCO-a has to be filtered to be relevant for the captions
  - encoder input structure might have to be changed
  - hyperparameters might have to be tuned more carefully
- would have been interesting to include image vector, but too little time
- evaluation other than manual is difficult

Lessons learned I

Organisation:

- technical Problems first, then the experiments
- deciding on data sets and input variants and creating the input in the right format takes a lot of time and requires good communication and planning
- share solutions (e.g., SLURM Tutorial update)
- time Plan for GPUs
- use parallel processes
- processes automation (bash scripts, framework for manual evaluation)

Lessons learned II

Practical skills

- build and improve a complex NN model for the special problem
- use Tensorflow, Git, GPUs
- tune hyperparameters
- create environments
- successfully generate grammatical sentences given textual attributes

Influence of different techniques / parameters:

- Attention; Word Embeddings; Dropout; Number of layers

Funny Examples

- A woman and a woman are feeding a baby to a man
- A man is sitting on a horse in a large room filled with other men
- Two people sitting at a table eating a dog
- A man standing in a kitchen with a big wooden baseball bear
- A woman feeding a giraffe to a giraffe at a zoo
- A man sitting on a couch with a red beard on his chest
- A group of people standing around a dead cake

References

[LMBBGHPRDZ14] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár und C. Lawrence Zitnick. „Microsoft COCO: Common Objects in Context". In: *CoRR* abs/1405.0312 (2014). arXiv: `1405.0312`. URL: `http://arxiv.org/abs/1405.0312`.

[RP15] Matteo Ruggero Ronchi und Pietro Perona. „Describing Common Human Visual Actions in Images". In: (2015). Hrsg. von Mark W. Jones Xianghua Xie und Gary K. L. Tam, S. 52.1–52.12.