Visual Re-ranking with Natural Language Understanding for Text Spotting

Ahmed Sabir[1], Francesc Moreno-Noguer[2], Lluís Padró[1]

[1] TALP Research Center, Universitat Politècnica de Catalunya, Barcelona, Spain

[2] Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain

ACCV
5.12.2018

Outline

Introduction

Motivation

Proposed Architecture

Dataset

Experiment

Conclusion

Outline

Introduction

Motivation

Proposed Architecture

Dataset

Experiment

Conclusion

Introduction

- Understanding the semantic relation between text and its environmental visual context show promising result in image information retrieval, such as object, location and logo retrieval

Images from Coco-text: Dataset and benchmark for text detection and recognition in natural images

Related work

Work addresses scene understanding, and benefit from combining text cue and visual context in image retrieval:

Text Detection Zhu et al.(2016)	semantic segmentation of **text background**
Lexicon Generation Patel et al.(2016)	generation of new lexicon with **topic modeling**
Logo Retrieval Karaoglu et al.(2017)	learn **textual information** from logos
Image Retrieval Bai X et al.(2017)	image retrieval with **text cue**

Related work

Work addresses scene understanding, and benefit from combining text cue and visual context in text retrieval:

Lexicon Generation
Patel et al.(2016)

generation of new lexicon with **topic modeling**

Logo Retrieval
Karaoglu et al.(2017)

learn **textual information** from logos

Image Retrieval
Bai X et al.(2017)

image retrieval with **text cue**

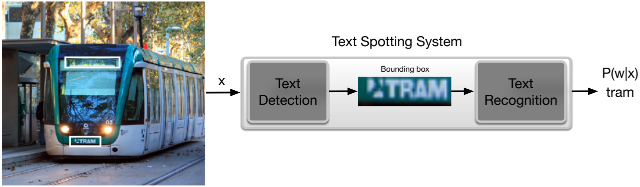
Text Retrieval
This work (2018)

enhance text spotting with **visual semantic**

What is Text Spotting?

End-to-End Text Recognition

- **Text Detection**: discover and locate the regions containing the text form natural images.
- **Text Recognition**: converting the detection text regions into computer readable material
- **Text Spotting**: an end-to-end text recognition system that accomplishes both tasks



Outline

Introduction

Motivation

Proposed Architecture

Dataset

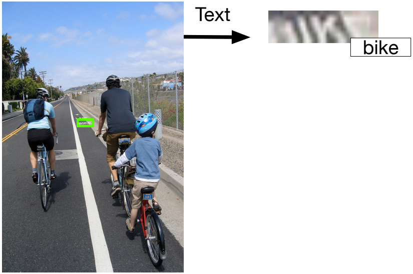
Experiment

Conclusion

Motivation

Goal

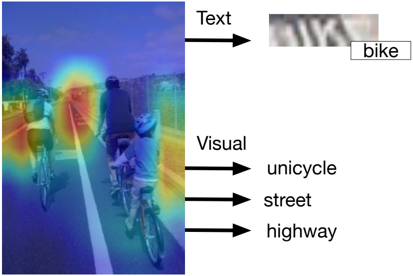
- Investigate the semantic relation between the text and the scene, and its influence on the accuracy.
- Propose a general approach that aims to fill the gap between Natural Language Understanding and vision in text spotting.



Motivation

Goal

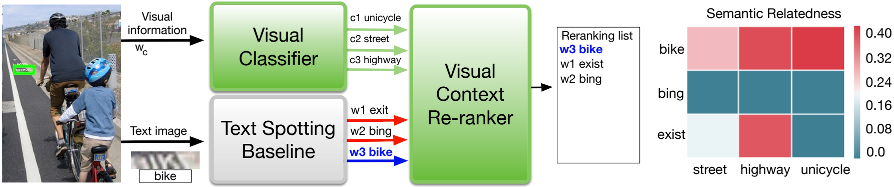
- Investigate the semantic relation between the text and the scene, and its influence on the accuracy.
- Propose a general approach that aims to fill the gap between Natural Language Understanding and vision in text spotting.



Motivation

Approach

- We propose a post-processing approach that intend to learn the semantic relation between the text and the scene.
- A simple scheme to improve the accuracy of any pre-trained text spotting algorithms without any computational cost.



Visual information w_c

Visual Classifier

c1 unicycle

c2 street

c3 highway

Text image

bike

Text Spotting Baseline

w1 exit

w2 bing

w3 bike

Visual Context Re-ranker

Reranking list

w3 bike

w1 exist

w2 bing

Outline

Introduction

Motivation

Proposed Architecture

Dataset

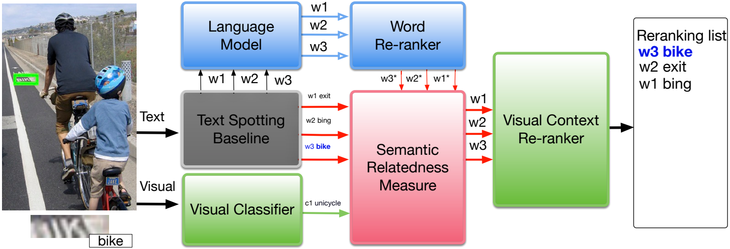
Experiment

Conclusion

Approach

Proposed Architecture

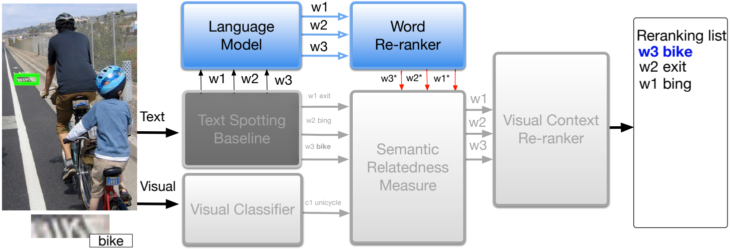
- Language Model (SLM, NLM)
- Semantic Relatedness Measure (word-embedding, NN, etc)
- Visual Classifier
- Visual Context Re-ranker



Approach

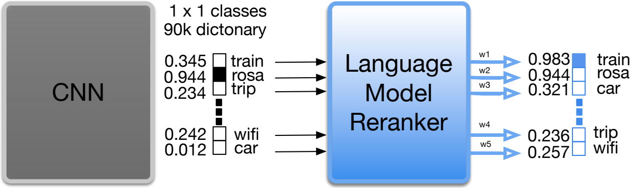
Proposed Architecture

- **Language Model**
- Semantic Relatedness Measure
- Visual Classifier
- Visual Context Re-ranker



Unigram Language Model

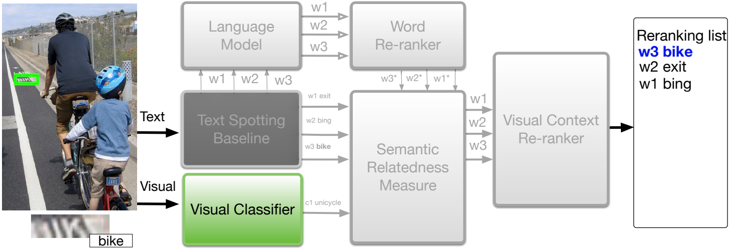
- The ULM is trained on a combined corpus (Opensubtitle and Google-book-ngram) (Lison and Tiedemann, 2016) **7M tokens**
- The advantage of ULM is very simple to build, train and adapt to new domains opening the possibility to improve baseline performance for specific applications.



Approach

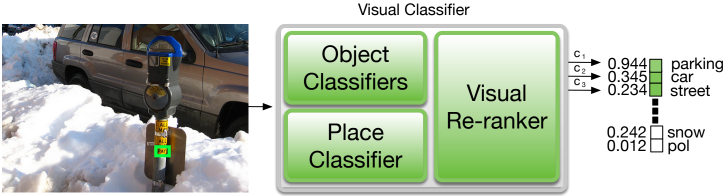
Proposed Architecture

- Language Model
- Semantic Relatedness Measure
- **Visual Classifier**
- Visual Context Re-ranker



Visual Classifier

- We consider four pre-trained object and scene classifiers to extract visual context information (Resnet152, Inception-v1, Inception-Resnet-v2, place365-resnet152)*
- The output of these classifier is a 1000 object instances.
- The output of the scene classifier is a 365 categories.
- We only consider the most likely objects-scene in the image by the classifier (k=3) with threshold (β) to filter out the probabilities prediction when the visual classifier not confident.

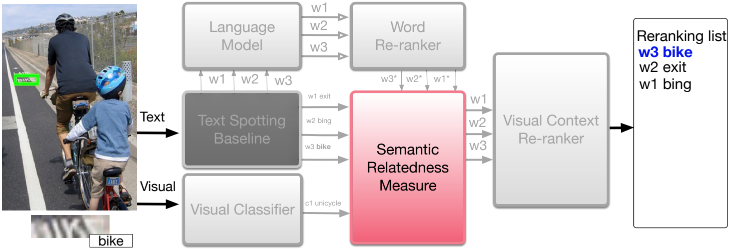


[*] please refer to the paper for all references

Approach

Proposed Architecture

- Language Model
- **Semantic Relatedness Measure**
- Visual Classifier
- Visual Context Re-ranker



Text

Visual

bike

Language Model

w1 w2 w3

Word Re-ranker

w3* w2* w1*

Text Spotting Baseline

w1 exit

w2 bing

w3 **bike**

Visual Classifier

c1 unicycle

Semantic Relatedness Measure

w1 w2 w3

Visual Context Re-ranker

Reranking list

w3 bike

w2 exit

w1 bing

Semantic Relatedness Measure I

- Word Embedding, **skip-gram** [1] trained on general text (SWE)

$$C_{max} = \quad \underset{\substack{c_i \in Image \\ P(c_i) \geq \beta}}{} sim(w, c_i)$$

$$sim(w, c) = \frac{\vec{w} \cdot \vec{c}}{|\vec{w}| \cdot |\vec{c}|}$$

- We convert the semantic score to probability according to assumption $p(w|c) \geqslant p(w)$ [2]. Thus the visual context asset the language model

$$P_{SWE}(w|c_{max}) = P(ULM)^{\alpha} \qquad \text{where } \alpha = \left(\frac{1-sim(w,c_{max})}{1+sim(w,c_{max})}\right)^{1-P(c_{max})}$$

- If there is no visual context information, we back-off to $\alpha = 1$ and use the bare unigram probability.

[1] Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality."NIPS. 2013.
[2] Blok, Sergey, Douglas Medin, and Daniel Osherson. "Probability from similarity." AAAI. 2003.

Semantic Relatedness Measure II

- Word Embedding, **skip-gram** [1] with negative sampling/NCE loss [3], trained on the dataset from scratch (TWE)

$$C_{max} = \quad \underset{\substack{c_i \in Image \\ P(c_i) \geq \beta}}{} sim(w, c_i)$$

$$sim(w, c) = \frac{\vec{w} \cdot \vec{c}}{|\vec{w}| \cdot |\vec{c}|}$$

- We convert the similarity to probability without the language model

$$P_{TWE}(w|c) = \frac{\tanh(sim(w, c)) + 1}{2P(c)}$$

[3] Mnih, Andriy, and Koray Kavukcuoglu. "Learning word embeddings efficiently with NCE." NIPS 2013.

Semantic Relatedness Measure III

- Estimating Relatedness from Training Day Probabilities (TDP)

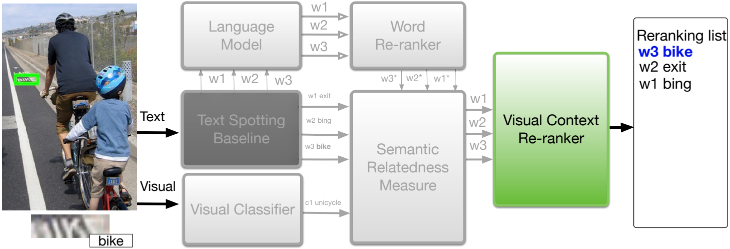
$$P_{TDP}(w|c) = \frac{count(w,c)}{count(c)}$$

- To overcome the cases of words not found in the embedding lexicon (e.g. commercial brands, quite common in images)

Approach

Proposed Architecture

- Language Model
- Semantic Relatedness Measure
- Visual Classifier
- **Visual Context Re-ranker**



Reranking list
w3 bike
w2 exit
w1 bing

Visual Context Re-ranker

Reranking Text Hypotheses (cascade)

- Semantic Relatedness with Word Embedding (SWE)

$$P_1(w, c) = P_{BL}(w) \times P_{SWE}(w|c)$$

- Estimating Relatedness from Training (TDE)

$$P_2(w, c) = P_{BL}(w) \times P_{TDP}(w|c)$$

- Semantic Relatedness with Word Embedding Revisited (TWE)

$$P_3(w, c) = P_{BL}(w) \times P_{TWE}(w|c)$$

Outline

Introduction

Motivation

Proposed Architecture

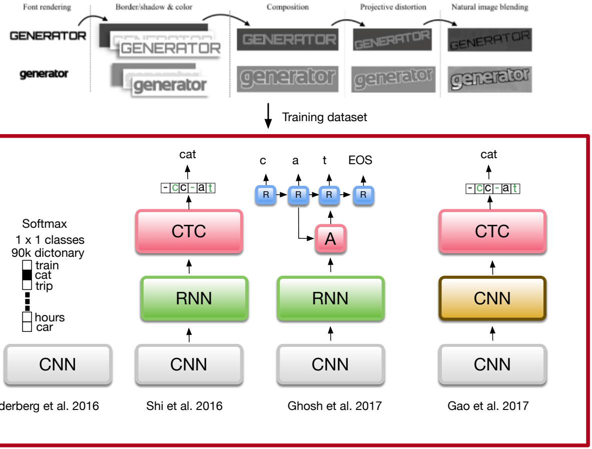
Dataset

Experiment

Conclusion

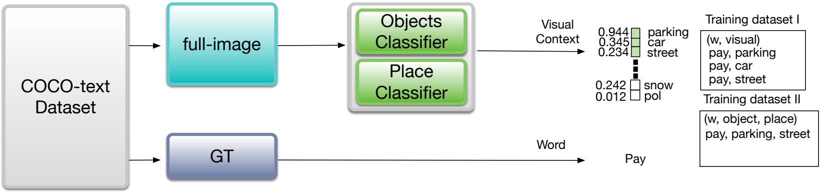
Dataset

- All current state-of-the-art are trainded on synthetic word dataset (Jaderberg et al. 2014)



Dataset - COCO-text

- COCO-text (Veit et al., 2016) is based on the MS COCO dataset, which contains images of complex everyday scenes (173,589 labeled text regions in over 63,686 images)
- Our dataset contains 15K full image with the bounding box and visual information (BBOx, $word_{gt}$, c_{places}, $c_{objects}$)
- For evaluation, we use ICDAR2017 Robust Reading Challenge on COCO-Text (end-to-end task).



Dataset is publicly available https://github.com/ahmedssabir/dataset/

Outline

Introduction

Motivation

Proposed Architecture

Dataset

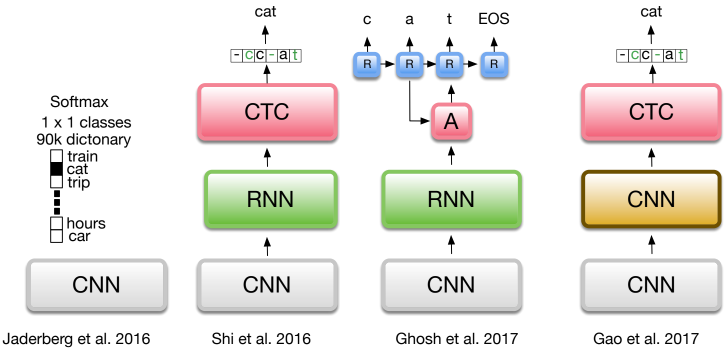
Experiment

Conclusion

Experiment

Baseline

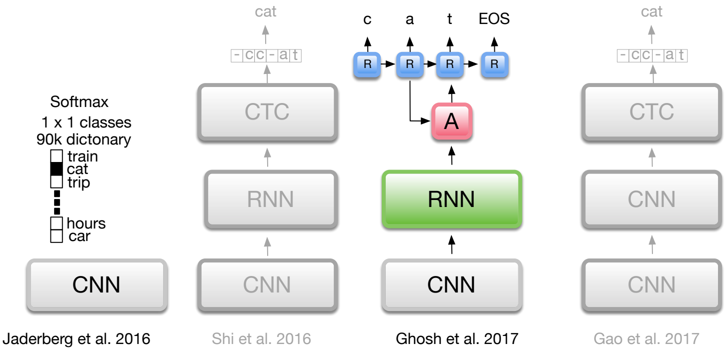
- CNN with 90K dictionary (fixed lexicon)
- LSTM with attention model (lexicon free)



Experiment

Baseline

- CNN with 90K dictionary (fixed lexicon)
- LSTM with attention model (lexicon free)



Result

- We evaluate all dataset (including word less than 3 characters and alphanumeric characters) unlike current protocol by state-of-the art.
- Simple example comparing all models :

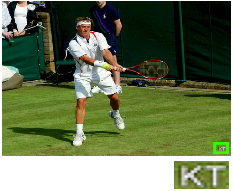
Word	Visual	SWE	TDP	TWE	TWE*
delta	airliner	0.0028	**0.0398**	0.0003	0.00029
kt	racket	0.0004	**0.0187**	0.0002	0.00006
plate	moving	0.0129	0.00050	**0.326**	0.00098
way	street	0.1740	0.02165	**0.177**	0.17493

Result

- We extract from k = 2 to 10 most likely words hypotheses –and their probabilities– from the baselines and re-rank theme using the Visual
- We able to improve both baselines 2%
- In case of the CNN, Dictionary 5.4%
- With CNN we able Retrieve 82.6% of the correct labels
- With LSTM we able to Retrieve 68.3% Lexicon-Free recognition

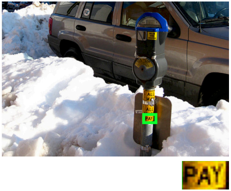
Model	**CNN**				**LSTM**		
	full	*dict*	*list*	*k*	*full*	*list*	*k*
Baseline	**full: 21.1 dict: 58.6**				**full: 18.7**		
TWE$_{TDP}$	23.0	64.0	75.2	9	20.8	68.3	9
SWE$_{TDP+objects}$	23.0	64.0	82.6	5	20.6	69.1	8
SWE$_{TDP+places}$	22.8	68.4	81.9	5	20.4	68.2	8
TWE$_{TDP}$ + SWE$_{TDP+places}$	22.8	63.4	82.1	5	20.3	72.9	5
TWE$_{TDP}$ + SWE$_{TDP+object}$	22.9	63.6	81.9	5	20.4	66.8	9

Result - Examples



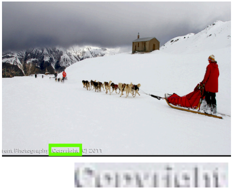
Reranking list:
w2: kt
w1: kr
w3: rt

Visual:
c1: racket
c2: grass



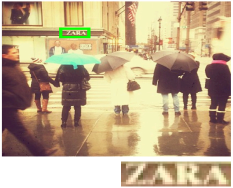
Reranking list:
w3: pay
w2: spay
w1: posy

Visual:
c1: parking
c2: igloo



Reranking list:
w1: convicting
w2: copyrighting
w3: cognizingly

Visual:
c1: ski slop
c2: snowfield



Reranking list:
w1: yard
w2: zara
w3: vara

Visual:
c1: crosswalk
c2: plaza

Outline

Introduction

Motivation

Proposed Architecture

Dataset

Experiment

Conclusion

Conclusion

Contributions

- We proposed a general architecture that, can be used as a **drop-in** replacement for any text-spotting algorithm that ranks the output words, uses semantic association to improve text recognition in images in the wild with low computational cost
- We re-defined the task of text spotting by exploring the semantic relation between text and scene. Also, introducing a visual context dataset for this problem.

Final thoughts

- Text in images is **not always related** to its visual environment, there is only a fraction of cases this approach may help solving, but given its low cost, it may be useful for domain adaptation of general text spotting systems.

Future work

- We plan to explore end-to-end fusion scheme that can automatically discover more proper priors in on one shot deep model fusion architecture.
- Add more visual context such as image description and sound
- Investigate the cases when visual context information is not useful for text spotting even from human perceptive.

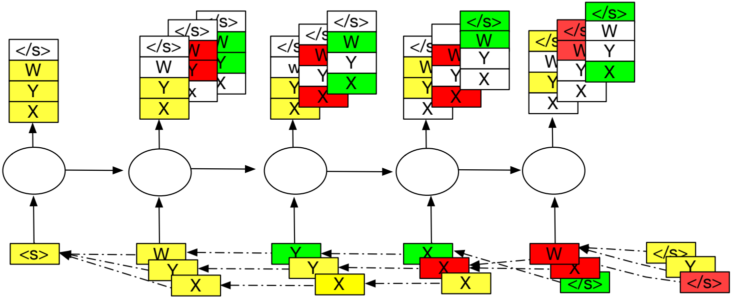
Thank You

Why post-processing ?

- **Lack of public dataset** (Most state-of-art deep models trained on synthetic dataset).
- **Fast and easy to re-train** Statistical Language Modelling (LM) can be trained on specific domain
- The system can be used as a **drop-in replacement** for any text-spotting algorithm that ranks the output words
- This **hybrid approach** between deep learning and classical statistical modelling opens the possibility to produce accurate results with very simple models.

LSTM and ULM

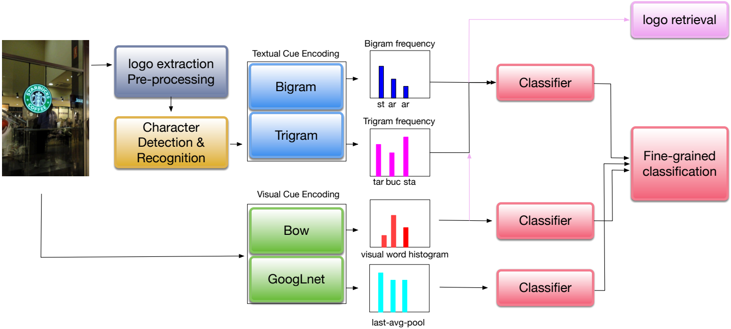
- Finding all possible combinations of all possible output words and choose a word (length 23 cha)
- Take the word with the highest probability (greedy)
- The highest probability goes to ULM



[Figure] Marcello Federico

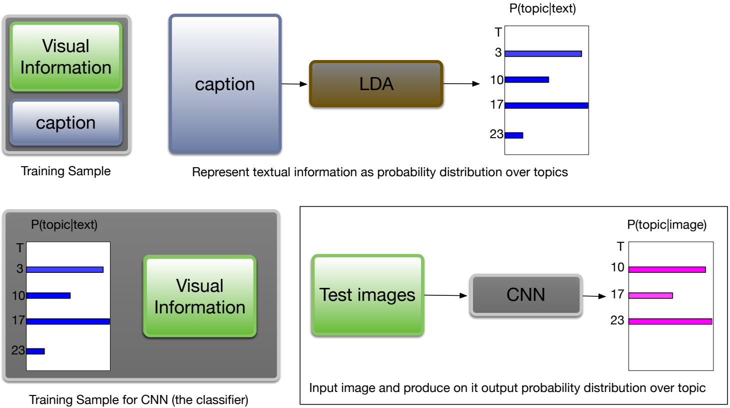
Related work

- The work of karaoglu el al.(2017) perform cue encoding Bigram and Trigram to propose the spatial pairwise reaction with the visual.
- Then, extracting visual cue for fine-grained classification.
- In short, this approach use textual information to distinguish between objects and logos.



Related work

- The work of Patel el al.(2016) use visual prior information to generate new lexicon.This approach use topic modelling (LDA) to learn the relation between text and images.



Training Sample

Represent textual information as probability distribution over topics

Training Sample for CNN (the classifier)

Input image and produce on it output probability distribution over topic