

Deep Joint-Semantics Reconstructing Hashing for Large-Scale Unsupervised Cross-Modal Retrieval

ICCV2019

*ShupengSu<sup>†</sup> ZhishengZhong<sup>†</sup> ChaoZhang<sup>\*</sup>*

Key Laboratory of Machine Perception (MOE), School of EECS, Peking University

Dejie Yang

University of China Academy of Science  
Insitute of Information Engineering, China Acadamey of Science

2019.12

# Outline

- 1 Motivation
- 2 Joint-Semantics Reconstructing Hashing
  - Joint-Semantics Matrix
  - Reconstructing with Binary Codes
- 3 Experiments
  - Datasets
  - Evaluation Criterion
  - Implementation Details
  - Retrieval Performance
  - Ablation Study
- 4 Think & Conclusion
- 5 Code

# Outline

- 1 Motivation
- 2 Joint-Semantics Reconstructing Hashing
  - Joint-Semantics Matrix
  - Reconstructing with Binary Codes
- 3 Experiments
  - Datasets
  - Evaluation Criterion
  - Implementation Details
  - Retrieval Performance
  - Ablation Study
- 4 Think & Conclusion
- 5 Code

# Motivation

- Here you can see an itemization
- Here you can see an itemization
- Here you can see an itemization
  - It has items
  - It has items
    - It has items
    - The items are below each other

# Outline

- 1 Motivation
- 2 Joint-Semantics Reconstructing Hashing
  - Joint-Semantics Matrix
  - Reconstructing with Binary Codes
- 3 Experiments
  - Datasets
  - Evaluation Criterion
  - Implementation Details
  - Retrieval Performance
  - Ablation Study
- 4 Think & Conclusion
- 5 Code

# Joint-Semantics Matrix

Top right of the slide shows subsections

# Reconstructing with Binary Codes

- ① Here you can see an enumeration
- ② It has items
- ③ The items are numbered

$$f(x) = \sum_{i=0}^{\infty} \frac{f^{(i)}(x_0)}{i!} (x - x_0)^i$$

# Outline

- 1 Motivation
- 2 Joint-Semantics Reconstructing Hashing
  - Joint-Semantics Matrix
  - Reconstructing with Binary Codes
- 3 Experiments
  - Datasets
  - Evaluation Criterion
  - Implementation Details
  - Retrieval Performance
  - Ablation Study
- 4 Think & Conclusion
- 5 Code



# Datasets

## Lorem Ipsum

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

## Observation

Simmons Dormitory is composed of brick.

## Evaluation Criterion

### Lorem Ipsum

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

### Observation

Simmons Dormitory is composed of brick.

## Implementation Details

### Lorem Ipsum

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

### Observation

Simmons Dormitory is composed of brick.

## Retrieval Performance

### Lorem Ipsum

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

### Observation

Simmons Dormitory is composed of brick.

## Ablation Study

### Lorem Ipsum

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

### Observation

Simmons Dormitory is composed of brick.

# Outline

- 1 Motivation
- 2 Joint-Semantics Reconstructing Hashing
  - Joint-Semantics Matrix
  - Reconstructing with Binary Codes
- 3 Experiments
  - Datasets
  - Evaluation Criterion
  - Implementation Details
  - Retrieval Performance
  - Ablation Study
- 4 Think & Conclusion**
- 5 Code

# Think & Conclusion

## Paragraph function

Write a function 'paragraph' that constructs a picture of width  $w$  of some text  $t$ , such that the content splits into as many lines as needed to fit into a paragraph of  $w$  columns.

## paragraph.ml

```
let paragraph s n =
  let rec traverse buffer n i = function
    | [] -> Picture.row (Lst.reverse buffer)
    | x::xs ->
      if i = n then traverse (x::'\n':buffer) n 1 xs
      else traverse (x::buffer) n (i+1) xs in
    traverse [] n 0 (Strng.of_string s);;
```

# Outline

- 1 Motivation
- 2 Joint-Semantics Reconstructing Hashing
  - Joint-Semantics Matrix
  - Reconstructing with Binary Codes
- 3 Experiments
  - Datasets
  - Evaluation Criterion
  - Implementation Details
  - Retrieval Performance
  - Ablation Study
- 4 Think & Conclusion
- 5 Code



## Github Code

### Paragraph function

Write a function 'paragraph' that constructs a picture of width  $w$  of some text  $t$ , such that the content splits into as many lines as needed to fit into a paragraph of  $w$  columns.

### paragraph.ml

```
let paragraph s n =
  let rec traverse buffer n i = function
    | [] -> Picture.row (Lst.reverse buffer)
    | x::xs ->
      if i = n then traverse (x::'\n'::buffer) n 1 xs
      else traverse (x::buffer) n (i+1) xs in
    traverse [] n 0 (Strng.of_string s);;
```

Thank you for your attention!