

Hypernym discovery : CRIM experiments(8) and Recurrent Mapping Model

CRIM, RMM 구현 및 실험

Geonil Yun,

rjsdlf45@gmail.com

2021. 11. 16.



Overview

- **CRIM – Hypernym discovery**
 - Euclidean distance
 - Use pre-trained embeddings
- **Recurrent Mapping Model – Hypernym discovery**
 - Model Structure
- **Scene Text**
 - TextFuseNet

CRIM – Hypernym Discovery

- Euclidean distance

Hyponym	Hypernym	Unsupervised	Supervised	Unsupervised - Pretrain
blackfly	homopterous insect	7.016	5.851	16.564
	insect	5.032	4.784	1.878
pollution	dirtiness	4.458	4.880	2.886
	dirtying	4.394	4.232	2.967
	environmental condition	1.795	3.715	9.074
song	possession	3.599	4.810	1.590
	sound	1.718	3.837	0.920
	work of art	2.030	3.296	10.447
	musical composition	2.544	3.609	15.002
	vocal music	3.212	4.772	14.203
	musical work	2.635	3.430	14.230
wing	emblem	2.322	3.948	1.177
	technical specification	3.529	3.430	14.355
	form	2.065	4.063	0.886
	specifications	3.644	3.682	2.000
Average		4.450	4.170	11.248

CRIM – Hypernym Discovery

- Pre-trained Embeddings

- Use pretrained embeddings

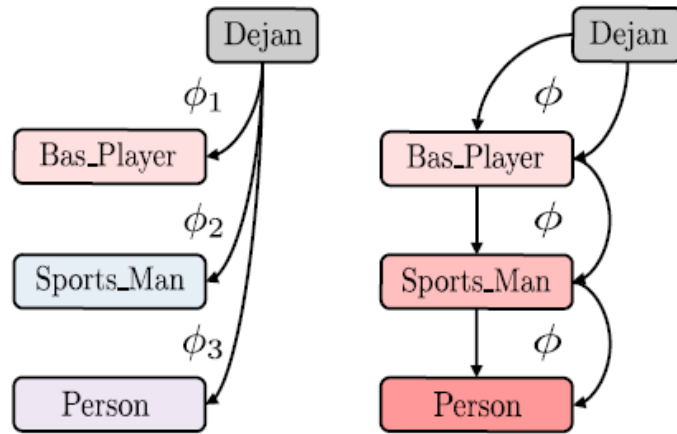
- Google-News embeddings (300 dimension) – experimenting now
 - Glove.6B embeddings (300 dimension)
 - **FastText** embeddings (300 dimension)



- Regard word as a **combination of characters**, embedding **n-gram characters** using skip-gram.
 - Embedding n-gram characters and the whole word.

Recurrent Mapping Models - Hypernym Discovery

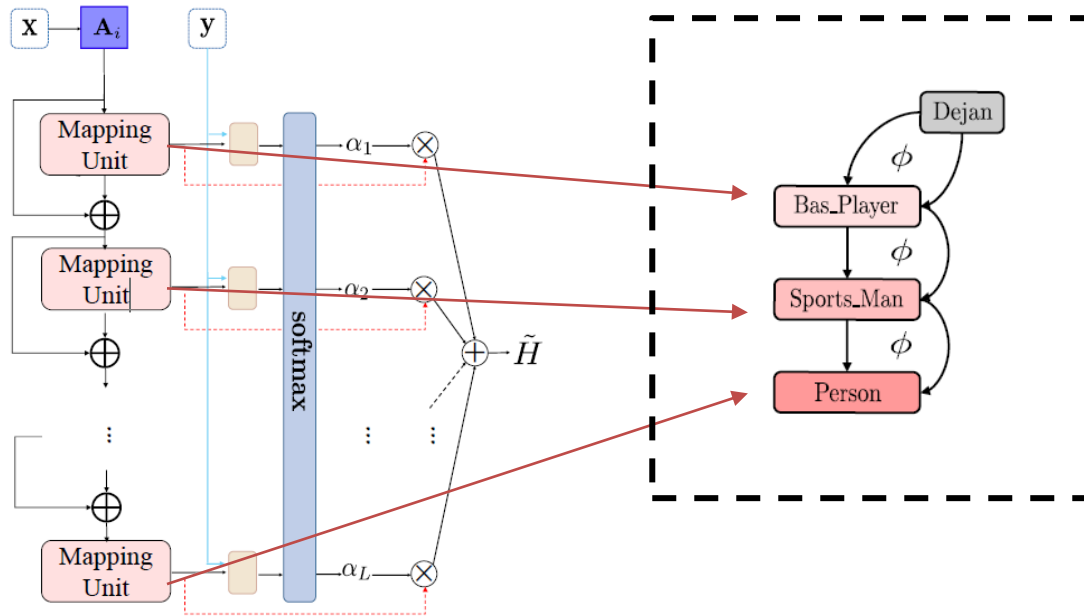
- RMM – Recurrent Mapping Models



- Assume each 'is-a' relation comes from **different level**.
- Higher-level hypernym transformed from previous-level hypernym.

Recurrent Mapping Models - Hypernym Discovery

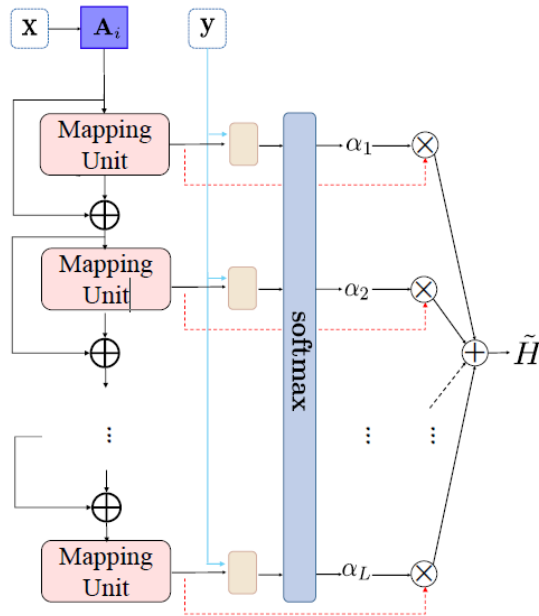
- Model Structure



- X : Embedding of query.
- y : Embedding of hypernym candidate.
- A_i : Transformation matrix ($i=2$, for Concept and Entity).
- Mapping Unit : Mapping terms to higher-level hypernym using projection matrices that share other mapping units.

Recurrent Mapping Models - Hypernym Discovery

- Model Structure



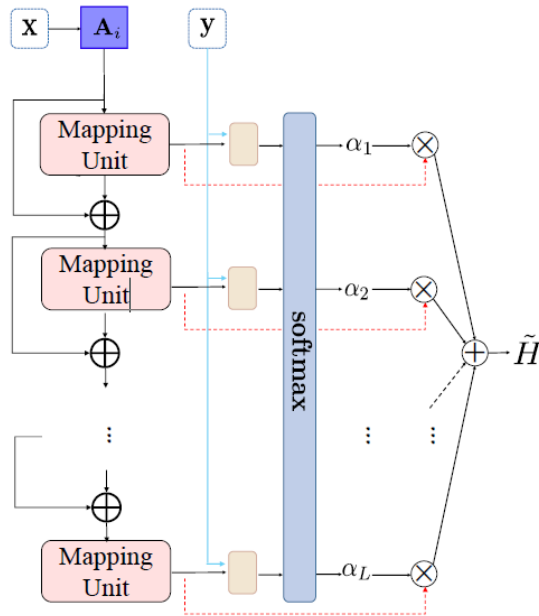
Type enhanced mapping and Recurrent Mapping Module

- $t = A_i X, t \in \mathbb{R}^{d \times 1}, A \in \mathbb{R}^{d \times d}$
- $\hat{h}_{l+1} = W_\phi h_l, h_0 = t, W_\phi \in \mathbb{R}^{d \times d}$
- $h_{l+1} = \hat{h}_{l+1} + h_l$

- Use **transform matrices** A_i for type enhanced representation t .
- Mapping unit map h_l to $l + 1^{th}$ level hypernym by using **projection matrix** W_ϕ .
- Using **residual connection** to overcome **gradient vanishing problem**.

Recurrent Mapping Models - Hypernym Discovery

- Model Structure



Aggregation Module

- $\alpha_l = \frac{\exp(h_l \cdot y)}{\sum_{k=1}^L \exp(h_k \cdot y)}$
- $\tilde{H} = \sum_{l=1}^L \alpha_l \cdot h_l$
- $s_y = \tilde{H} \cdot y$

- Get **attention weight** of a level.
- Each level's attention score is probability of the candidate hypernym **lying in that level**.

Issue & Future

- last week

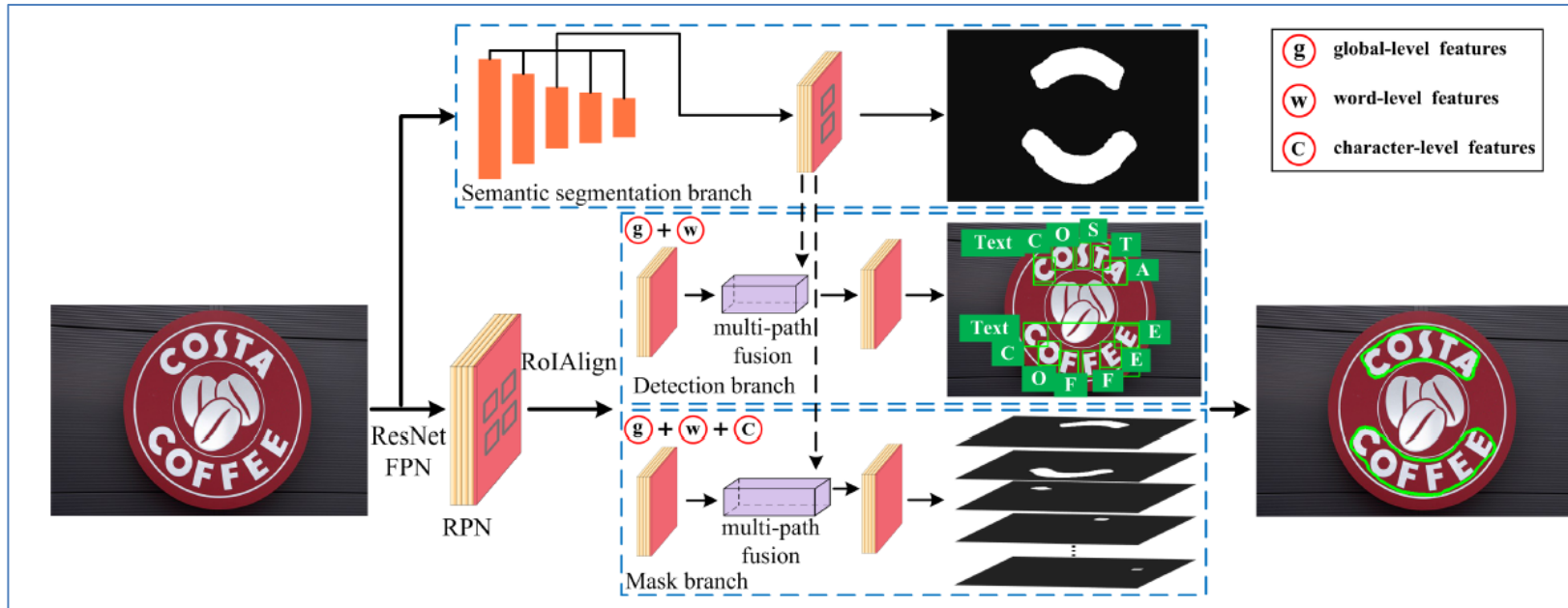
- RMM(Recurrent Mapping Model)
- Use pre-trained embeddings.
- More experiments.

- Future works

- RMM(Recurrent Mapping Model) - Loss function.
- Use other pre-trained embeddings.
- Other task dataset – hypernym detection (BLESS, EVAL ..)

Scene Text

- TextFuseNet



- Arbitrary shape text detection in images.
- Collect **text's features from different levels** using **multi-path fusion** structure.

Scene Text

- TextFuseNet

- Existing methods

- ① **Character-based** : regard the text as a **combination of multiple characters**.
⇒ **time-consuming** due to large number of character candidates generated for text detection.
- ② **Word-based** : directly detect words based on **generic object detection**.
⇒ usually failed to detect.

To improve this issue, **apply instance segmentation**. but, still two limitations. (**single ROI** and **level of word semantics**)

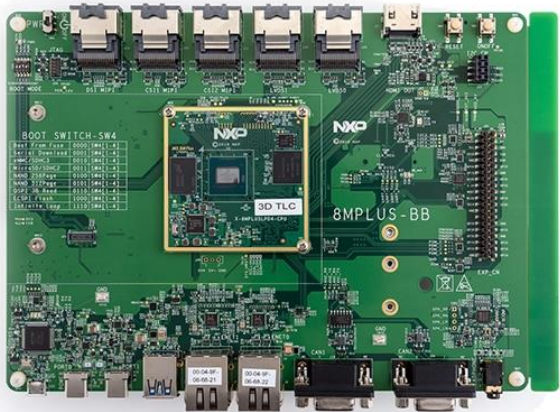


Scene Text

- **TextFuseNet**
 - Follow **Mask R-CNN**, **Mask TextSpotter**.
 - **Recast** original Mask R-CNN pipeline.
 - Formulate the text detection task as an **instance segmentation task**.
 - Fusion 3-level of features(**character**, **word**, **global**).

Scene Text

- NXP Board : 8M PLUS LPD4 - EVK
 - Specifications



Processor	4x Arm Cortex-A53 1.8Gz
	1x Arm Cortex-M7 800MHz
	HiFi4 DSP 800MHz
	Nueral Processing Unit
Memory	LPDDR4 6GB
	32GB eMMC 5.1
	32MB QSPI NOR

Thank you