LIAAD at SemDeep-5 Challenge

Word-in-Context (WiC)

Daniel Loureiro, Alípio Jorge

SemDeep-5@IJCAI – Macau, 12 August 2019
Sense Embeddings

Exploiting the latest Neural Language Models (NLMs) for sense-level representation learning.
Sense Embeddings

Exploiting the latest Neural Language Models (NLMs) for sense-level representation learning.

- Beat SOTA for English Word Sense Disambiguation (WSD).
- Full WordNet in NLM-space (+100K common sense concepts).
- Concept-level analysis of NLMs. [ACL 2019 – LMMS Paper]
Related Work
Related Work

Bag-of-Features Classifiers
(SVM)

Deep Sequence Classifiers
(BiLSTM)

Sense-level Representations
(k-NN)
(over NLM reprs.)

[Iacobacci et al. (2016)]
[Zhong and Ng (2010)]

[Luo et al. (2018b)]
[Luo et al. (2018a)]
[Vial et al. (2018)]
[Raganato et al. (2017)]

[Loureiro and Jorge (2019)]
[Peters et al. (2018)]
[Melamud et al. (2016)]
[Yuan et al. (2016)]
Contextual k-NN

Matching Contextual Word Embeddings:

• Produce Sense Embeddings from NLMs (averaging).
• Sense embs. can be compared with contextual embs.
• Disambiguation = Nearest Neighbour search (1-NN).
• Annotations have limited coverage (16% of WordNet).
• Promising, but early attempts.

[Introduction][Related Work][Our Approach][Performance][Conclusions]
Our Approach
Our Approach

- Expand the k-NN approach to full-coverage of WordNet.
Our Approach

• Expand the k-NN approach to full-coverage of WordNet.
• Matching senses becomes trivial, no MFS fallbacks needed.
Our Approach

• Expand the k-NN approach to full-coverage of WordNet.
• Matching senses becomes trivial, no MFS fallbacks needed.
• Full-set of sense embeddings in NLM-space is useful beyond WSD.
Our Approach

- Expand the k-NN approach to full-coverage of WordNet.
- Matching senses becomes trivial, no MFS fallbacks needed.
- Full-set of sense embeddings in NLM-space is useful beyond WSD.
Our Approach

- Expand the k-NN approach to full-coverage of WordNet.
- Matching senses becomes trivial, no MFS fallbacks needed.
- Full-set of sense embeddings in NLM-space is useful beyond WSD.
Challenges
Challenges

• Overcome very limited sense annotations (covers 16% senses).
• Infer missing senses correctly so that task performance improves.
• Rely only on sense embeddings, no lemma or POS features.*

*Covered on our ACL 2019 Paper
Can your insurance company aid you in reducing administrative costs?

Would it be feasible to limit the menu in order to reduce feeding costs?
Bootstrapping Sense Embeddings

Can your insurance company aid you in reducing administrative costs?

Would it be feasible to limit the menu in order to reduce feeding costs?

Introduction  Related Work  **Our Approach**  Performance  Conclusions
Bootstrapping Sense Embeddings

Can your insurance company aid you in reducing administrative costs?

Would it be feasible to limit the menu in order to reduce feeding costs?
Can your insurance company aid you in reducing administrative costs?

Would it be feasible to limit the menu in order to reduce feeding costs?
Bootstrapping Sense Embeddings

Introduction

Related Work

Our Approach

Performance

Conclusions
Bootstrapping Sense Embeddings

\[ \vec{c_1} \text{reduce%2:30:00::} \]
\[ \vec{c_2} \text{reduce%2:30:00::} \]
\[ \vec{c_1} \text{cost%1:21:00::} \]
\[ \vec{c_2} \text{cost%1:21:00::} \]

Introduction  Related Work  Our Approach  Performance  Conclusions
Bootstrapping Sense Embeddings

\[ \vec{v}_{\text{reduce2:30:00::}} = \frac{\vec{c}_1_{\text{reduce2:30:00::}} + \vec{c}_2_{\text{reduce2:30:00::}} + \ldots + \vec{c}_n_{\text{reduce2:30:00::}}}{n} \]

\[ \vec{v}_{\text{cost1:21:00::}} = \frac{\vec{c}_1_{\text{cost1:21:00::}} + \vec{c}_2_{\text{cost1:21:00::}} + \ldots + \vec{c}_n_{\text{cost1:21:00::}}}{n} \]
Bootstrapping Sense Embeddings

\[
\vec{v}_{\text{reduce%2:30:00::}} = \frac{\vec{c}_1_{\text{reduce%2:30:00::}} + \vec{c}_2_{\text{reduce%2:30:00::}} + \ldots + \vec{c}_n_{\text{reduce%2:30:00::}}}{n}
\]

\[
\vec{v}_{\text{cost%1:21:00::}} = \frac{\vec{c}_1_{\text{cost%1:21:00::}} + \vec{c}_2_{\text{cost%1:21:00::}} + \ldots + \vec{c}_n_{\text{cost%1:21:00::}}}{n}
\]

Outcome: 33,360 sense embeddings (16% coverage)
Propagating Sense Embeddings

WordNet’s units, synsets, represent concepts at different levels.
Propagating Sense Embeddings

WordNet’s units, synsets, represent concepts at different levels.
Propagating Sense Embeddings

WordNet’s units, synsets, represent concepts at different levels.
Propagating Sense Embeddings

burger%1:13:00::
hotdog%1:18:00::
hamburger%1:13:01::
sandwich%1:13:00::
wrap%1:13:00::
potato_chip%1:13:00::

Introduction  Related Work  Our Approach  Performance  Conclusions
Propagating Sense Embeddings

Introduction
Related Work
Our Approach
Performance
Conclusions
Propagating Sense Embeddings

Retrieve Synsets, Relations and Categories

noun.food

sandwich.n.01

burger.n.02  hotdog.n.01  wrap.n.02  chips.n.04

Introduction  Related Work  Our Approach  Performance  Conclusions
Propagating Sense Embeddings

1st stage: Synset Embeddings

noun.food

sandwich.n.01

burger.n.02  hotdog.n.01  wrap.n.02  chips.n.04

burger%1:13:00::  hamburger%1:13:01::  sandwich%1:13:00::  wrap%1:13::00  potato_chip%1:13::00

hotdog%1:18::00

Introduction  Related Work  Our Approach  Performance  Conclusions
Propagating Sense Embeddings

2\textsuperscript{nd} Stage: Hypernym Embeddings (ind. Synsets)

noun.food

sandwich.n.01

burger.n.02 hotdog.n.01 wrap.n.02 chips.n.04

Introduction Related Work Our Approach Performance Conclusions
Propagating Sense Embeddings

3rd Stage: Lexname Embeddings

noun.food

sandwich.n.01

burger.n.02 hotdog.n.01 wrap.n.02 chips.n.04

burger%1:13:00:: hamburger%1:13:01:: sandwich%1:13:00:: wrap%1:13::00 potato_chip%1:13::00

hotdog%1:18::00

Introduction Related Work Our Approach Performance Conclusions
Propagating Sense Embeddings

But 🍔 != 🌯 ...

noun.food

sandwich.n.01

burger.n.02 hotdog.n.01 wrap.n.02 chips.n.04

burger%1:13:00:: hamburger%1:13:01:: sandwich%1:13:00:: wrap%1:13:00 potato_chips%1:13:00

hotdog%1:18:00

Introduction Related Work Our Approach Performance Conclusions
Enriching Sense Embeddings

Leverage Synset Definitions and Lemmas for Differentiation
Enriching Sense Embeddings

Leverage Synset Definitions and Lemmas for Differentiation

sandwich:%1:13:00:: (sandwich.n.01)
Definition: two (or more) slices of bread with a filling between them
Lemmas: sandwich

wrap:%1:13:00:: (wrap.n.02)
Definition: a sandwich in which the filling is rolled up in a soft tortilla
Lemmas: wrap, tortilla

Introduction  Related Work  Our Approach  Performance  Conclusions
Enriching Sense Embeddings

Compose a new context

sandwich:%1:13:00:: (sandwich.n.01)
sandwich - two (or more) slices of bread with a filling between them

wrap:%1:13:00:: (wrap.n.02)
wrap, tortilla - a sandwich in which the filling is rolled up in a soft tortilla
Enriching Sense Embeddings

Make the context specific to sensekey (repeat lemma)

sandwich:%1:13:00::
sandwich - sandwich - two (or more) slices of bread with a filling between them

wrap%1:13:00::
wrap - wrap, tortilla - a sandwich in which the filling is rolled up in a soft tortilla
Enriching Sense Embeddings

Make the context specific to sensekey (repeat lemma)

sandwich:%1:13:00::
sandwich - sandwich - two (or more) slices of bread with a filling between them

wrap%1:13:00::
wrap - wrap, tortilla - a sandwich in which the filling is rolled up in a soft tortilla
Enriching Sense Embeddings

Obtain contextual embeddings for every token

sandwich:%1:13:00::
sandwich - sandwich - two (or more) slices of bread with a filling between them

wrap%1:13:00::
wrap – wrap, tortilla - a sandwich in which the filling is rolled up in a soft tortilla
Enriching Sense Embeddings

Sentence Embedding from avg. of Contextual Embeddings

sandwich: \%1:13:00::

\[ \overrightarrow{v_d} = \] sandwich: \%1:13:00::

wrap\%1:13:00::

\[ \overrightarrow{v_d} = \] wrap\%1:13:00::

d = 1024
Enriching Sense Embeddings

Merge Sentence Embedding with previous Sense Embedding

\[
\overrightarrow{v_d} = \text{sandwich} \quad \overrightarrow{v_s} = \text{sandwich} \\
\overrightarrow{v_d} = \text{wrap} \quad \overrightarrow{v_s} = \text{wrap}
\]
Enriching Sense Embeddings

Merge Sentence Embedding with previous Sense Embedding

$$\vec{v}_s =$$

sandwich:%1:13:00::

$$d = 2048$$

wrap%1:13:00::
Matching Sense Embeddings

The glasses are in the cupboard.
Matching Sense Embeddings

The glasses are in the cupboard.
Matching Sense Embeddings

\[ \vec{v}_t = \begin{bmatrix} \vec{c} \\ \vec{\hat{c}} \end{bmatrix} \]

The glasses are in the cupboard.
Matching Sense Embeddings

The glasses are in the cupboard.
WSD Performance
Standard English WSD Evaluation
F1 on ALL set of the WSD Evaluation Framework (Raganato et al. 2017)
Classifying Embedding Similarities

Sentence Tokens: Marco makes ravioli | Apple makes iPhones

Contextual Embeddings:

Sense Embeddings:
Classifying Embedding Similarities

Sentence Tokens: Marco makes ravioli Apple makes iPhones

Contextual Embeddings: [Visual representation of embedding similarities]

Sense Embeddings: [Visual representation of embedding similarities]
Classifying Embedding Similarities

Sentence Tokens: Marco makes ravioli Apple makes iPhones

Contextual Embeddings:

Sense Embeddings:
(cook.v.02) (produce.v.02)

Introduction Related Work Our Approach Performance Conclusions
# Classifying Embedding Similarities

<table>
<thead>
<tr>
<th>Sentence Tokens:</th>
<th>Marco</th>
<th>makes</th>
<th>ravioli</th>
<th>Apple</th>
<th>makes</th>
<th>iPhones</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contextual Embeddings:</td>
<td><img src="image" alt="Contextual Embeddings" /></td>
<td><img src="image" alt="Contextual Embeddings" /></td>
<td><img src="image" alt="Contextual Embeddings" /></td>
<td><img src="image" alt="Contextual Embeddings" /></td>
<td><img src="image" alt="Contextual Embeddings" /></td>
<td><img src="image" alt="Contextual Embeddings" /></td>
</tr>
<tr>
<td>Sense Embeddings:</td>
<td><img src="image" alt="Sense Embeddings" /></td>
<td><img src="image" alt="Sense Embeddings" /></td>
<td><img src="image" alt="Sense Embeddings" /></td>
<td><img src="image" alt="Sense Embeddings" /></td>
<td><img src="image" alt="Sense Embeddings" /></td>
<td><img src="image" alt="Sense Embeddings" /></td>
</tr>
</tbody>
</table>

(cook.v.02) (produce.v.02)
Classifying Embedding Similarities

Sentence Tokens: Marco makes ravioli Apple makes iPhones

Contextual Embeddings:

Sense Embeddings:
(cook.v.02) (produce.v.02)
Classifying Embedding Similarities

Sentence Tokens: Marco makes ravioli Apple makes iPhones

Contextual Embeddings:

Sense Embeddings:
(cook.v.02) sim₃ (produce.v.02) sim₄

Now, we classify different similarity combinations using Binary Logistic Regression

Introduction Related Work Our Approach Performance Conclusions
Classifying Embedding Similarities

Sentence Tokens: Marco makes ravioli Apple makes iPhones

Contextual Embeddings:

Sense Embeddings:

(cook.v.02) (produce.v.02)

Baseline M0 WSD (no train/dev) M1 sim1 M2 sim2 M3 sim1, 2 M4 sim1, 2, 3, 4 SuperGlue

Introduction Related Work Our Approach Performance Conclusions
Conclusions

• Systems designed for WSD, without being trained for the WiC task, can perform competitively.

• Sense Embeddings can still benefit from information captured by contextual embeddings, as shown by similarities classifier.

• In future work, progress on the WiC task could lead to better semi-supervised annotations for WSD.
Thanks

Code and Sense Embeddings: github.com/danlou/LMMS

dloureiro@fc.up.pt  @danielbloureiro