Regularized and Retrofitted models for Learning Sentence Representation with Context

Tanay Kumar Saha\textsuperscript{1} \hspace{1em} Shafiq Joty\textsuperscript{2} \hspace{1em} Naeemul Hassan\textsuperscript{3} \\
Mohammad Al Hasan\textsuperscript{1}

\textsuperscript{1}Indiana University Purdue University Indianapolis, Indianapolis, IN 46202, USA

\textsuperscript{2}Nanyang Technological University, Singapore

\textsuperscript{3}University of Mississippi, Oxford, Mississippi

November 7, 2017
Outline

1. Introduction and Motivation
2. Our Approach for Learning Sentence Representation
3. Experiments
4. Conclusion
1. Introduction and Motivation
   - Distributed Representation of Sentences
   - Motivation

2. Our Approach for Learning Sentence Representation
   - Content Model
   - Context Types
   - Regularized Models
   - Retrofitted Models

3. Experiments
   - Evaluation Tasks and Datasets
   - Classification and Clustering Performance
   - Summarization Performance

4. Conclusion
Represent sentences with **condensed real-valued vectors** that capture syntactic and semantic properties of the sentence

- *I play soccer* \(\Rightarrow [0.2, 0.3, 0.4]\)

Many sentence-level text processing tasks rely on representing sentences with fixed-length vectors

- The most common approach uses bag-of-ngrams (e.g., tf.idf)

Distributed representation has been shown to perform better
Motivation

- Most existing Sen2Vec methods disregard **context** of a sentence
- Meaning of one sentence depends on the meaning of its neighbors
  - And I was wondering about the GD LEV
  - *Is it reusable?*
  - *Or is it discarded to burn up on return to LEO?*
- Our approach: incorporate **extra-sentential context** into Sen2Vec
- We propose two methods: **regularization** and **retrofitting**
- We experiment with two types of context: **discourse** and **similarity**.
1. Introduction and Motivation
   - Distributed Representation of Sentences
   - Motivation

2. Our Approach for Learning Sentence Representation
   - Content Model
   - Context Types
   - Regularized Models
   - Retrofitted Models

3. Experiments
   - Evaluation Tasks and Datasets
   - Classification and Clustering Performance
   - Summarization Performance

4. Conclusion
Our Approach

- Consider **content** as well as **context** of a sentence
- Treat the context sentences as **atomic** linguistic units
  - Similar in spirit to (Le & Mikolov, 2014)
  - Efficient to train compared to **compositional** methods like encoder-decoder models (e.g., SDAE, Skip-Thought)
Content Model (Sen2Vec)

- Treats sentences and words similarly
- Represented by vectors in shared embedding matrix
- \( \mathbf{v} \): he works in woodworking

\[ \phi : V \rightarrow \mathbb{R}^d \]

Figure: Distributed bag of words or DBOW (Le & Mikolov, 2014)
Context Types

- **Discourse Context**
  - Formed by *previous* and *following* sentences in the text
  - Adjacent sentences in a text are logically connected by certain coherence relations (e.g., elaboration, contrast)

- **Similarity Context**
  - Based on more *direct measures* of similarity (e.g., cosine)
  - Considers similarity with all other sentences

- Context can be represented by a **graph neighborhood**, \( \mathcal{N}(v) \)
Similarity Network Construction

- Represent the sentences with vectors learned from Sen2Vec, then measure the cosine similarity between the vectors.
- Restrict context size of a sentence for computational efficiency.
- Set thresholds for intra- and across-document connections.
- Allow up to 20 most similar neighbors.
Regularized Models (Reg-dis, Reg-sim)

- Incorporate neighborhood directly into the objective function of the content-based model (Sen2Vec) as a regularizer.

- Objective function:

\[
J(\phi) = \sum_{v \in V} \left[ \mathcal{L}_c(v) + \beta \mathcal{L}_r(v, N(v)) \right]
\]

\[
= \sum_{v \in V} \left[ \mathcal{L}_c(v) + \beta \sum_{(v, u) \in E} \| \phi(u) - \phi(v) \|^2 \right] \quad (1)
\]

- Train with SGD
- Regularization with discourse context ⇒ Reg-dis
- Regularization with similarity context ⇒ Reg-sim
u: And I was wondering about the GD LEV.

v: Is it reusable?

y: Or is it discarded to burn up on return to LEO?

(a) A sequence of sentences

(b) Sen2Vec (DBOW)

(c) REG-DIS

φ

υ

L_r

L_r
Retrofitted Model (RET-dis, RET-sim)

- Retrofit vectors learned from Sen2Vec s.t. the revised vector $\phi(v)$:
  - Similar to the prior vector, $\phi'(v)$
  - Similar to the vectors of its neighboring sentences, $\phi(u)$

- Objective function:
  
  $$ J(\phi) = \sum_{v \in V} \alpha_v \|\phi(v) - \phi'(v)\|^2 + \sum_{(v,u) \in E} \beta_{u,v} \|\phi(u) - \phi(v)\|^2 $$

- Solve using Jacobi iterative method
- Retrofit with discourse context $\Rightarrow$ RET-dis
- Retrofit with similarity context $\Rightarrow$ RET-sim
### 1. Introduction and Motivation
- Distributed Representation of Sentences
- Motivation

### 2. Our Approach for Learning Sentence Representation
- Content Model
- Context Types
- Regularized Models
- Retrofitted Models

### 3. Experiments
- Evaluation Tasks and Datasets
- Classification and Clustering Performance
- Summarization Performance

### 4. Conclusion
### Extractive summarization (ranking task)

- Select the most important sentences to form a summary
- Use the popular graph-based algorithm LexRank
  - nodes $\Rightarrow$ sentences
  - edges $\Rightarrow$ cosine similarity between vectors (learned by models)
- Benchmark datasets from **DUC-01** and **DUC-02** for evaluation

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Doc.</th>
<th>#Avg. Sen.</th>
<th>#Avg. Sum.</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUC 2001</td>
<td>486</td>
<td>40</td>
<td>2.17</td>
</tr>
<tr>
<td>DUC 2002</td>
<td>471</td>
<td>28</td>
<td>2.04</td>
</tr>
</tbody>
</table>
Evaluation Tasks and Datasets

1. **Topic classification and clustering**

   - Use learned vectors to classify or cluster sentences into topics
   - MaxEnt classifier and K-means++ clustering algorithm
   - Text categorization corpora: **Reuters-21578** & **20-Newsgroups**.

   - But, we need sentence-level annotation for evaluation
   - Naive assumption: sentences of a document share the same topic label as the document $\Rightarrow$ induces lot of noise
   - Our approach: LexRank to select top 20% sentences of each document as representatives of the document

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Doc.</th>
<th>Total # sen.</th>
<th>Annot. # sen</th>
<th>Train # sen</th>
<th>Test # sen</th>
<th>#Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reuters</td>
<td>9,001</td>
<td>42,192</td>
<td>13,305</td>
<td>7,738</td>
<td>3,618</td>
<td>8</td>
</tr>
<tr>
<td>Newsgroups</td>
<td>7,781</td>
<td>95,809</td>
<td>22,374</td>
<td>10,594</td>
<td>9,075</td>
<td>8</td>
</tr>
</tbody>
</table>
Classification and Clustering Performance

<table>
<thead>
<tr>
<th>Topic Classification Results</th>
<th>Topic Clustering Results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table:** Performance on topic classification & clustering in comparison to Sen2Vec
<table>
<thead>
<tr>
<th>Method</th>
<th>DUC’01</th>
<th>DUC’02</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sen2Vec</td>
<td>43.88</td>
<td>54.01</td>
</tr>
<tr>
<td>Tf-Idf</td>
<td>(+) 4.83</td>
<td>(+) 1.51</td>
</tr>
<tr>
<td>W2V-avg</td>
<td>(-) 0.62</td>
<td>(+) 1.44</td>
</tr>
<tr>
<td>C-PHRASE</td>
<td>(+) 2.52</td>
<td>(+) 1.68</td>
</tr>
<tr>
<td>FastSent</td>
<td>(-) 4.15</td>
<td>(-) 7.53</td>
</tr>
<tr>
<td>Skip-Thought</td>
<td>(+) 0.88</td>
<td>(-) 2.65</td>
</tr>
<tr>
<td>RET-sim</td>
<td>(-) 0.62</td>
<td>(+) 0.42</td>
</tr>
<tr>
<td>RET-dis</td>
<td>(+) 0.45</td>
<td>(-) 0.37</td>
</tr>
<tr>
<td>REG-sim</td>
<td>(+) 2.90</td>
<td>(+) 2.02</td>
</tr>
<tr>
<td>REG-dis</td>
<td>(-) 1.92</td>
<td>(-) 8.77</td>
</tr>
</tbody>
</table>

**Table:** ROUGE-1 scores on DUC datasets in comparison to Sen2Vec
Outline

1. Introduction and Motivation
   - Distributed Representation of Sentences
   - Motivation

2. Our Approach for Learning Sentence Representation
   - Content Model
   - Context Types
   - Regularized Models
   - Retrofitted Models

3. Experiments
   - Evaluation Tasks and Datasets
   - Classification and Clustering Performance
   - Summarization Performance

4. Conclusion
Conclusion and Future Work

► Novel models for learning vector representation of sentences that consider not only content of a sentence but also its context
► Two ways to incorporate context: retrofitting and regularizing
► Two types of context: discourse and similarity
► Discourse context beneficial for topic classification and clustering, whereas the similarity context beneficial for summarization
► Explore further how our models perform compared to existing compositional models, where documents with sentence-level sentiment annotation exists
Thanks!

- Code and Datasets: https://github.com/tksaha/con-s2v/tree/jointlearning
- Check our CON-S2V ECML-2017 paper