Lesson 10
Deep learning for NLP:
Multilingual Word Sequence Modeling

December 15, 2016

EPFL Doctoral Course EE-724
Nikolaos Pappas
Idiap Research Institute, Martigny
Outline of the talk

1. Recap: Word Representation Learning

2. Multilingual Word Representations
   • Alignment models
   • Evaluation tasks

3. Multilingual Word Sequence Modeling
   • Essentials: RNN, LSTM, GRU
   • Machine Translation
   • Document Classification

4. Summary

* Figure from Lebret's thesis, EPFL, 2016
Disclaimer

- Research highlights rather than in-depth analysis
  - By no means exhaustive (progress too fast!)
  - Tried to keep most representatives
- Focus on feature learning and two major NLP tasks
- Not enough time to cover other exciting tasks:
  - Question answering
  - Relation classification
  - Paraphrase detection
  - Summarization
Recap: Learning word representations from text

- Why should we care about them?
  - tackles curse of dimensionality
  - captures semantic and analogy relations of words
  - captures general knowledge in an unsupervised way

\[
\text{king} - \text{man} + \text{woman} \approx \text{queen}
\]
Recap: Learning word representations from text

• **How can we benefit from them?**
  • study linguistic properties of words
  • inject general knowledge on downstream tasks
  • transfer knowledge across languages or modalities
  • compose representations of word sequences
Recap: Learning word representations from text

- **Which method to use for learning them?**
  - neural versus count-based methods
    - neural ones implicitly do SVD over a PMI matrix
    - similar to count-based when using the same tricks
  - neural methods appear to have the edge (word2vec)
    - efficient and scalable objective + toolkit
    - intuitive formulation (=predict words in context)
Recap: Continuous Bag-of-Words (CBOW)
Recap: Continuous Bag-of-Words (CBOW)

**Positive:** he sat on a **chair**

**Negative:** he sat on a **oxygen**

\[ W_h \in \mathbb{R}^{k \cdot d \times h} \]
\[ W_{out} \in \mathbb{R}^{h \times |V|} \]

Maximize \((0, 1 - s + s_c)\)

Back propagate and update word representations

Advantage: does not require this expensive matrix multiplication
Recap: Learning word representations from text

- What else can we do with word embeddings?
  - dependency-based embeddings: Levy and Goldberg 2014
  - retrofitted-to-lexicons embeddings: Faruqui et al. 2014
  - sense-aware embeddings: Li and Jurafsky 2015
  - visually-grounded embeddings: Lazaridou et al. 2015
  - multilingual embeddings: Gouws et al. 2015
Outline of the talk

1. Recap: Word Representation Learning

2. Multilingual Word Representations
   - Alignment models
   - Evaluation tasks

3. Multilingual Word Sequence Modeling
   - Essentials: RNN, LSTM, GRU
   - Machine Translation
   - Document Classification

4. Summary

* Figure from Gouts et al., 2015.
Learning cross-lingual word representations

- Monolingual embeddings capture semantic, syntactic and analogy relations between words
- **Goal**: capture this relationships two or more languages

* Figure from Gouts et al., 2015.
Supervision of cross-lingual alignment methods

- Parallel sentences for MT: Guo et al., 2015
  Sentence by sentence and word alignments

- Parallel sentences: Gouws et al., 2015
  Sentence by sentence alignments

- Parallel documents: Søgaard et al., 2015
  Documents with topic or label alignments

- Bilingual dictionary: Ammar et al., 2016
  Word by word translations

- No parallel data: Faruqui and Dyer, 2014
  Really!

Annotation cost

- high
- low
Cross-lingual alignment with no parallel data

After Stage 1: \( X = \) representations of English words
\( Y = \) representations of French words

Goal: transform \( X \) and \( Y \) such that the transformed representations of (cat, chat), (you, toi), etc. are close to each other

maximize \( \text{trace}(A^T X^T Y^T B) \)
\[ \text{s.t. } A^T X^T X A = I \]
\[ B^T Y^T Y B = I \]

(Faruqui and Dyer, 2014)
Cross-lingual alignment with parallel sentences

Training data: Parallel sentences

- $a =$ English sentence
- $b =$ parallel French sentence
- $n =$ random French sentence

minimize

$$E(a, b) = \|f(a) - g(b)\|^2$$

minimize

$$\max(0, m + E(a, b) - E(a, n))$$

Compose word representations to get a sentence representation using a Compositional Vector Model (CVM)

Two options considered:

ADD: (simply add word vectors)

$$s = \text{sentence}$$

$$w_i = \text{representation of word } i \text{ in the sentence}$$

$$f(s) = \sum_{i=1}^{n} w_i$$

BI (gram):

$$f(s) = \sum_{i=1}^{n} \tanh(w_{i-1} + w_i)$$

(Hermann & Blunson, 2014)
Cross-lingual alignment with parallel sentences

\[
\text{Fr positive: } Il \text{ était assis sur une } \text{chaise} \\
\text{Fr negative: } Il \text{ était assis sur une } \text{oxygène}
\]

\[
\text{En positive: he sat on a } \text{chair} \\
\text{En negative: he sat on a } \text{oxygen}
\]

Independently update \( \theta^e \) and \( \theta^f \)

\[
\text{maximize } \max(0, 1 - s^f + s^f_c) \quad \text{w.r.t. } \theta^e
\]

+ Parallel data

\[
\text{En: he sat on a chair } [s_e = w_1^e, w_2^e, w_3^e, w_4^e, w_5^e] \\
\text{Fr: Il était assis sur une chaise } [s_f = w_1^f, w_2^f, w_3^f, w_4^f, w_5^f]
\]

now, also minimize \( \Omega(W_{emb}^e, W_{emb}^f) = \| \frac{1}{m} \sum_{w_i \in s^e} W_{emb}^e - \frac{1}{n} \sum_{w_j \in s^e} W_{emb}^f \|^2 \)

w.r.t \( W_{emb}^e, W_{emb}^f \)

maximize \( \max(0, 1 - s^e + s^e_c) \) \text{ w.r.t. } \theta^f

(Gouws et al., 2015)
Cross-lingual alignment with parallel sentences for MT

**English Training instance:** he sat on a chair

In addition also update French words in proportional to their similarity to \{he, sat, on, a\}

(Klementiev et. al., 2012)
Unified framework for analysis of cross-lingual methods

- Minimize monolingual objective
- Constraint/Regularize with bilingual objective

\[
\text{maximize } \sum_{j \in \{e,f\}} \sum_{i=1}^{T_j} \omega_j \left( \frac{\mathcal{L}(\theta_j)}{\log n} \right) + \lambda \cdot \Omega (W^e_{emb}, W^f_{emb})
\]

\[
\begin{aligned}
&\text{w.r.t } \theta_e, \theta_f \\
&\theta_e = W_e, W^e_h, W^e_{out} \\
&\theta_f = W_f, W^f_h, W^f_{out}
\end{aligned}
\]

monolingual similarity

bilingual similarity

Nikolaos Pappas
Evaluation: Cross-lingual document classification and translation

<table>
<thead>
<tr>
<th>Method</th>
<th>en $\rightarrow$ de</th>
<th>de $\rightarrow$ en</th>
<th>Training Time (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Majority Baseline</td>
<td>46.8</td>
<td>46.8</td>
<td>-</td>
</tr>
<tr>
<td>Glossed Baseline</td>
<td>65.1</td>
<td>68.6</td>
<td>-</td>
</tr>
<tr>
<td>MT Baseline</td>
<td>68.1</td>
<td>67.4</td>
<td>-</td>
</tr>
<tr>
<td>Klementiev et al.</td>
<td>77.6</td>
<td>71.1</td>
<td>14,400</td>
</tr>
<tr>
<td>Bilingual Auto-encoders (BAEs)</td>
<td>91.8</td>
<td>72.8</td>
<td>4,800</td>
</tr>
<tr>
<td>BiCVM</td>
<td>83.7</td>
<td>71.4</td>
<td>15</td>
</tr>
<tr>
<td>BilBOWA (this work)</td>
<td>86.5</td>
<td>75</td>
<td>6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>En$\rightarrow$Sp P@1</th>
<th>Sp$\rightarrow$En P@1</th>
<th>En$\rightarrow$Sp P@5</th>
<th>Sp$\rightarrow$En P@5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Distance</td>
<td>13</td>
<td>18</td>
<td>24</td>
<td>27</td>
</tr>
<tr>
<td>Word Co-occurrence</td>
<td>30</td>
<td>19</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Mikolov et al., 2013</td>
<td>33</td>
<td>35</td>
<td>51</td>
<td>52</td>
</tr>
<tr>
<td>BilBOWA (this work)</td>
<td>39 (+6)</td>
<td>44 (+9)</td>
<td>51</td>
<td>55 (+3)</td>
</tr>
</tbody>
</table>

(Gows et al., 2015)
Bonus: Multilingual visual sentiment concept matching

concept = adjective-noun-phrase (ANP)

(Pappas et al., 2016)
Multilingual visual sentiment concept ontology

- 7.36M+ Flickr images
- ~16K affective visual concepts: Adjective-Noun Pairs (ANPs)
- Co-occurrence (emotion, ANP)
- Sentiment value (text-based)
- 12 languages detected

<table>
<thead>
<tr>
<th>Language</th>
<th>Concepts</th>
<th>Images</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>4421</td>
<td>447997</td>
</tr>
<tr>
<td>Spanish</td>
<td>3381</td>
<td>37528</td>
</tr>
<tr>
<td>Italian</td>
<td>3349</td>
<td>25664</td>
</tr>
<tr>
<td>French</td>
<td>2349</td>
<td>16807</td>
</tr>
<tr>
<td>Chinese</td>
<td>504</td>
<td>5562</td>
</tr>
<tr>
<td>German</td>
<td>804</td>
<td>7335</td>
</tr>
<tr>
<td>Dutch</td>
<td>348</td>
<td>2226</td>
</tr>
<tr>
<td>Russian</td>
<td>129</td>
<td>800</td>
</tr>
<tr>
<td>Turkish</td>
<td>231</td>
<td>638</td>
</tr>
<tr>
<td>Polish</td>
<td>63</td>
<td>477</td>
</tr>
<tr>
<td>Persian</td>
<td>15</td>
<td>34</td>
</tr>
<tr>
<td>Arabic</td>
<td>29</td>
<td>23</td>
</tr>
</tbody>
</table>

(Jou et al., 2015)
Word embedding model

- Skip-gram model (word2vec)\(^1\)
  - Google News 100B
  - Wikipedia 1.74B
  - Wikipedia + Reuters + WSJ 1.96B
  - Flickr 100 Million 0.75B
- Concept vectors
  - Sum of words composition
  - Directly learned (ANPs as tokens)

---

\(^1\) Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado and Jeffrey Dean
*Distributed Representations of Words and Phrases and their Compositionality*
NIPS, Lake Tahoe, Nevada, USA, 2013

(Pappas et al., 2016)
Multilingual visual sentiment concept retrieval

- How often do two visual concepts appear together?
  - Tag co-occurrence matrix \((n \times n)\)
- ANPs can be described as
  - Co-occurrence vectors \(h_i, h_j\) in \(R^n\)
    - \(n\) is the number of translated ANPs

- Visual semantic distance between ANPs

\[
d(ANP_i, ANP_j) = 1 - \cos(h_i, h_j)
\]

(Pappas et al., 2016)
Multilingual visual sentiment concept clustering

Visual **Semantic** Relatedness for different clustering methods

For each clustering method:

\[
\text{sem}_C = \frac{1}{C} \sum_{c=1}^{C} \frac{\sum_{j : j \neq i \& U_{ij} \neq 0} d(\text{ANP}_{c,i}, \text{ANP}_{c,j})}{N_c}
\]

- **Average over all clusters**
- Average visual semantic distance in a cluster for all ANP pairs whose semantic distance is greater than 0
- \(C\) = number of non-unary clusters
- \(N_c\) = number of ANPs for a cluster \(c\)

(Pappas et al., 2016)
Multilingual visual sentiment concept clustering

Visual **Sentiment** Consistency for different clustering methods

For each clustering method:

$$\text{sen}_C = \frac{1}{C} \sum_{c=1}^{C} \left( \frac{\sum_{i=1}^{N_c} \left( \text{sen}(\text{ANP}_{c,i}) - \text{sen}_c \right)^2}{N_c} \right)$$

- **C** = number of non-unary clusters
- **Nc** = number of ANPs for a cluster **c**

(Pappas et al., 2016)
Discovering interesting clusters: Multilingual

(Pappas et al., 2016)
Discovering interesting clusters: Western vs. Eastern

(Pappas et al., 2016)
Discovering interesting clusters: Monolingual

SPANISH
políticos corruptos
(corrupt politicians)

ITALIAN
carnevale ambrosiano
(ambrosian carnival)

FRENCH
travailleurs pauvres
(poor workers)

CHINESE
传统 灯笼
(traditional lantern)

ARABIC
قضية انسانية
(humanitarian issue)

(Pappas et al., 2016)
Evaluation: Multilingual visual sentiment concept analysis

- Aligned embeddings are better than translation in concept retrieval, clustering and sentiment prediction

<table>
<thead>
<tr>
<th>Method</th>
<th>Language</th>
<th>EN</th>
<th>ES</th>
<th>IT</th>
<th>FR</th>
<th>ZH</th>
<th>DE</th>
<th>NL</th>
<th>RU</th>
<th>TR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Translated concepts</td>
<td>w=5</td>
<td>5.94</td>
<td>4.86</td>
<td>5.49</td>
<td>5.23</td>
<td>5.41</td>
<td>6.27</td>
<td>7.96</td>
<td>13.50</td>
<td>11.72</td>
</tr>
<tr>
<td>Aligned concepts</td>
<td>w=5</td>
<td>5.94</td>
<td>3.05</td>
<td>3.77</td>
<td>4.20</td>
<td>2.22</td>
<td>4.08</td>
<td>6.60</td>
<td>17.83</td>
<td>15.85</td>
</tr>
<tr>
<td>Improvement (%)</td>
<td></td>
<td>+0.0</td>
<td>+59.3</td>
<td>+45.6</td>
<td>+24.5</td>
<td>+143.6</td>
<td>+53.6</td>
<td>+20.6</td>
<td>-32.0</td>
<td>-35.2</td>
</tr>
</tbody>
</table>

![Graphs showing sentiment consistency and accuracy](image)
Conclusion

• Aligned embeddings are cheaper than translation and usually work better than it in several multilingual or crosslingual NLP tasks without parallel data
  
  • document classification Gows et al., 2015
  
  • named entity recognition Al-Rfou et al., 2014
  
  • dependency parsing Guo et al., 2015
  
  • concept retrieval and clustering Pappas et al., 2016
Outline of the talk

1. Recap: Word Representation Learning

2. Multilingual Word Representations
   - Alignment models
   - Evaluation tasks

3. Multilingual Word Sequence Modeling
   - Essentials: RNN, LSTM, GRU
   - Machine Translation
   - Document Classification

4. Summary

* Figure from Colah’s blog, 2015.
Language Modeling

• Computes the probability of a sequence of words or simply “likelihood of a text”: \( P(w_1, w_2, \ldots, w_t) \)

• N-gram models with Markov assumption:

\[
P(w_1, \ldots, w_m) = \prod_{i=1}^{m} P(w_i \mid w_1, \ldots, w_{i-1}) \approx \prod_{i=1}^{m} P(w_i \mid w_{i-(n-1)}, \ldots, w_{i-1})
\]

• Where is it useful?
  • speech recognition
  • machine translation
  • POS tagging and parsing

• What are its limitations?
  • unrealistic assumption
  • huge memory needs
  • back-off models
Recurrent Neural Network (RNN)

• Neural language model:

\[
\hat{P}(x_{t+1} = v_j \mid x_t, \ldots, x_1) = \hat{y}_t = \text{softmax} \left( W^{(S)} h_t \right)
\]

\[
h_t = \sigma \left( W^{(hh)} h_{t-1} + W^{(hx)} x_t \right)
\]

• What are its main limitations?
  • vanishing gradient problem (error doesn’t propagate far)
  • fail to capture long-term dependencies
  • tricks: gradient clipping, identity initialization + ReLus
Long Short Term Memory (LSTM)

- Long-short term memory nets are able to learn long-term dependencies: Hochreiter and Schmidhuber 1997

Simple RNN:

* Figure from Colah’s blog, 2015.
Long Short Term Memory (LSTM)

- Long-short term memory nets are able to learn long-term dependencies: Hochreiter and Schmidhuber 1997
- Ability to remove or add information to the cell state regulated by “gates”

* Figure from Colah’s blog, 2015.
Gated Recurrent Unit (GRU)

• Gated RNN by Chung et al, 2014 combines the forget and input gates into a single “update gate”
  • keep memories to capture long-term dependencies
  • allow error messages to flow at different strengths

\[
\begin{align*}
  z_t &= \sigma (W_z \cdot [h_{t-1}, x_t]) \\
  r_t &= \sigma (W_r \cdot [h_{t-1}, x_t]) \\
  \tilde{h}_t &= \tanh (W \cdot [r_t \ast h_{t-1}, x_t]) \\
  h_t &= (1 - z_t) \ast h_{t-1} + z_t \ast \tilde{h}_t
\end{align*}
\]

zt: update gate — rt: reset gate — ht: regular RNN update

* Figure from Colah’s blog, 2015.
Deep Bidirectional Models

• Here RNN but it applies to LSTMs and GRUs too

\[
\begin{align*}
\vec{h}_t &= f(\overrightarrow{W} h_t^{(i-1)} + \overrightarrow{V} h_{t-1} + b^{(i)}) \\
\overleftarrow{h}_t &= f(\overleftarrow{W} h_t^{(i-1)} + \overleftarrow{V} h_{t+1} + b^{(i)}) \\
y_t &= g(U[h_t^{(L)} ; h_t^{(L)}] + c)
\end{align*}
\]

(Irsoy and Cardie, 2014)

Each memory layer passes an intermediate sequential representation to the next.
Convolutional Neural Network (CNN)

- Typically good for images
- Convolutional filter(s) is (are) applied every k words:
  \[ c_i = f(w^T x_{i:i+h-1} + b) \]
- Similar to Recursive NNs but without constraining to grammatical phrases only, as Socher et al., 2011
  - no need for a parser (!)
  - less linguistically motivated?

(Collobert et al., 2011)
(Kim, 2014)
Hierarchical Models

• Word-level and sentence-level modeling with any type of NN layers

(Tang et al., 2015)
Attention Mechanism for Machine Translation

• Chooses “where to look” or learns to assign a relevance to each input position given encoder hidden state for that position and the previous decoder state

• learns a soft bilingual alignment model

(Bahdanau et al., 2015)
Attention Mechanism for Document Classification

• Operates on input word sequence (or intermediate hidden states: Pappas and Popescu-Belis 2016)
• Learns to focus on relevant parts of the input with respect to the target labels
  • learns a soft extractive summarization model

(Pappas and Popescu-Belis, 2014)
Outline of the talk

1. Recap: Word Representation Learning

2. Multilingual Word Representations
   - Alignment models
   - Evaluation tasks

3. Multilingual Word Sequence Modeling
   - Essentials: RNN, LSTM, GRU
   - Machine Translation
   - Document Classification

4. Summary

* Figure from Colah’s blog, 2015.
RNN encoder-decoder for Machine Translation

- GRU as hidden layer
- Maximize the log likelihood of the target sequence given the source sequence:

$$\max_{\theta} \frac{1}{N} \sum_{n=1}^{N} \log p_{\theta}(Y_n | X_n)$$

- WMT 2014 (EN→FR)

![Diagram of RNN Encoder-Decoder](image)

Figure 1: An illustration of the proposed RNN Encoder–Decoder.

(Cho et al., 2014)
Sequence to sequence learning for Machine Translation

- LSTM hidden layers instead of GRU
- 4 layers deep instead of shallow encoder-decoder

(Sutskever et al., 2014)
Sequence to sequence learning for Machine Translation

- WMT 2014 (EN→FR)

<table>
<thead>
<tr>
<th>Method</th>
<th>test BLEU score (ntst14)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bahdanau et al. [2]</td>
<td>28.45</td>
</tr>
<tr>
<td>Baseline System [29]</td>
<td>33.30</td>
</tr>
<tr>
<td>Single forward LSTM, beam size 12</td>
<td>26.17</td>
</tr>
<tr>
<td>Single reversed LSTM, beam size 12</td>
<td>30.59</td>
</tr>
<tr>
<td>Ensemble of 5 reversed LSTMs, beam size 1</td>
<td>33.00</td>
</tr>
<tr>
<td>Ensemble of 2 reversed LSTMs, beam size 12</td>
<td>33.27</td>
</tr>
<tr>
<td>Ensemble of 5 reversed LSTMs, beam size 2</td>
<td>34.50</td>
</tr>
<tr>
<td>Ensemble of 5 reversed LSTMs, beam size 12</td>
<td>34.81</td>
</tr>
</tbody>
</table>

- PCA projection of the hidden state of the last encoder layer

(Sutskever et al., 2014)
Jointly learning to align and translate for Machine Translation

- **Limitation:** can we compress all the needed information in the last encoder state?
- **Idea:** use all the hidden states of the encoder
  - length proportional to that of the sentence!
  - compute a weighted average of all the hidden states

(Bahdanau et al., 2015)
Jointly learning to align and translate for Machine Translation

- WMT 2014 (EN→FR)

(Bahdanau et al., 2015)
Effective approaches to attention-based NMT

- Global and local attention
- Input-feeding approach
- Stacked LSTM instead of single-layer

(Luong et al., 2015)
Multi-source NMT

- Train $p(e | f, g)$ model directly on trilingual data
- Use it to decode $e$ given any $(f, g)$ pair
- Take local-attention NMT model and concatenate context from multiple sources

(Zoph and Knight, 2016)
Multi-source NMT

- Multi-source training improves over individual French English and German English pairs
  - Best: basic concatenation with attention

<table>
<thead>
<tr>
<th>Target = English</th>
<th>Source</th>
<th>Method</th>
<th>Ppl</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>French</td>
<td>—</td>
<td>10.3</td>
<td>21.0</td>
<td></td>
</tr>
<tr>
<td>German</td>
<td>—</td>
<td>15.9</td>
<td>17.3</td>
<td></td>
</tr>
<tr>
<td>French+German</td>
<td>Basic</td>
<td>8.7</td>
<td>23.2</td>
<td></td>
</tr>
<tr>
<td>French+German</td>
<td>Child-Sum</td>
<td>9.0</td>
<td>22.5</td>
<td></td>
</tr>
<tr>
<td>French+French</td>
<td>Child-Sum</td>
<td>10.9</td>
<td>20.7</td>
<td></td>
</tr>
<tr>
<td>French</td>
<td>Attention</td>
<td>8.1</td>
<td>25.2</td>
<td></td>
</tr>
<tr>
<td>French+German</td>
<td>B-Attent.</td>
<td>5.7</td>
<td>30.0</td>
<td></td>
</tr>
<tr>
<td>French+German</td>
<td>CS-Attent.</td>
<td>6.0</td>
<td>29.6</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Target = German</th>
<th>Source</th>
<th>Method</th>
<th>Ppl</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>French</td>
<td>—</td>
<td>12.3</td>
<td>10.6</td>
<td></td>
</tr>
<tr>
<td>English</td>
<td>—</td>
<td>9.6</td>
<td>13.4</td>
<td></td>
</tr>
<tr>
<td>French+English</td>
<td>Basic</td>
<td>9.1</td>
<td>14.5</td>
<td></td>
</tr>
<tr>
<td>French+English</td>
<td>Child-Sum</td>
<td>9.5</td>
<td>14.4</td>
<td></td>
</tr>
<tr>
<td>English</td>
<td>Attention</td>
<td>7.3</td>
<td>17.6</td>
<td></td>
</tr>
<tr>
<td>French+English</td>
<td>B-Attent.</td>
<td>6.9</td>
<td>18.6</td>
<td></td>
</tr>
<tr>
<td>French+English</td>
<td>CS-Attent.</td>
<td>7.1</td>
<td>18.2</td>
<td></td>
</tr>
</tbody>
</table>

(Zoph and Knight, 2016)
Multi-source NMT

- Multi-source training improves over individual French English and German English pairs
  - **Best**: basic concatenation with attention

---

<table>
<thead>
<tr>
<th>Source</th>
<th>Method</th>
<th>Ppl</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>French</td>
<td>—</td>
<td>10.3</td>
<td>21.0</td>
</tr>
<tr>
<td>German</td>
<td>—</td>
<td>15.9</td>
<td>17.3</td>
</tr>
<tr>
<td>French+German</td>
<td>Basic</td>
<td>8.7</td>
<td>23.2</td>
</tr>
<tr>
<td>French+German</td>
<td>Child-Sum</td>
<td>9.0</td>
<td>22.5</td>
</tr>
<tr>
<td>French+French</td>
<td>Child-Sum</td>
<td>10.9</td>
<td>20.7</td>
</tr>
<tr>
<td>French</td>
<td>Attention</td>
<td>8.1</td>
<td>25.2</td>
</tr>
<tr>
<td>French+German</td>
<td>B-Attent.</td>
<td>5.7</td>
<td>30.0</td>
</tr>
<tr>
<td>French+German</td>
<td>CS-Attent.</td>
<td>6.0</td>
<td>29.6</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Source</th>
<th>Method</th>
<th>Ppl</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>French</td>
<td>—</td>
<td>12.3</td>
<td>10.6</td>
</tr>
<tr>
<td>English</td>
<td>—</td>
<td>9.6</td>
<td>13.4</td>
</tr>
<tr>
<td>French+English</td>
<td>Basic</td>
<td>9.1</td>
<td>14.5</td>
</tr>
<tr>
<td>French+English</td>
<td>Child-Sum</td>
<td>9.5</td>
<td>14.4</td>
</tr>
<tr>
<td>English</td>
<td>Attention</td>
<td>7.3</td>
<td>17.6</td>
</tr>
<tr>
<td>French+English</td>
<td>B-Attent.</td>
<td>6.9</td>
<td>18.6</td>
</tr>
<tr>
<td>French+English</td>
<td>CS-Attent.</td>
<td>7.1</td>
<td>18.2</td>
</tr>
</tbody>
</table>

(Zoph and Knight, 2016)
Multi-target NMT

• Multi-task learning framework for multiple target language translation
  • Optimization for one to many model

(Dong et al., 2015)
Multi-target NMT

- Improves over NMT and moses baselines over WMT 2013 test
  - but also on larger datasets
- Faster and better convergence in multiple language translation

(Dong et al., 2015)
Multi-way, Multilingual NMT

- Encoder-decoder model with multiple encoders and decoders shared across pairs
  - share knowledge across langs
  - universal space for all langs
  - good for low-resource langs
- Attention is pair specific, hence expensive \(O(L^2)\)
  - instead share attention across all pairs!

**Figure:** n_th encoder and m_th decoder at timestep t / \(\phi\) makes encoder & decoder states compatible with the attention mechanism / \(f_{\text{adp}}\) makes context vector compatible with the decoder → all these transformations to support different types of encoders/decoders for different languages!

(Firat et al., 2016)
## Multi-way, Multilingual NMT

- **Consistent improvements for low-resource languages**
  - the lower the training data the bigger the improvement
- **In large-scale translation improves only translation to English**
  - **hypothesis**: EN appears always as source or target language for all pairs → better decoder?

### Consistent improvements for low-resource languages

- En→Fi
  - Size: 100k
    - Single: 5.06/3.96
    - Single+DF: 4.98/3.99
    - Multi: 6.2/5.17
  - Size: 200k
    - Single: 7.1/6.16
    - Single+DF: 7.21/6.17
    - Multi: 8.84/7.53
  - Size: 400k
    - Single: 9.11/7.85
    - Single+DF: 9.31/8.18
    - Multi: 11.09/9.98
  - Size: 800k
    - Single: 11.08/9.96
    - Single+DF: 11.59/10.15
    - Multi: 12.73/11.28

- En→De
  - Size: 210k
    - Single: 14.27/13.2
    - Single+DF: 14.65/13.88
    - Multi: 16.96/16.26
  - Size: 420k
    - Single: 18.32/17.32
    - Single+DF: 18.51/17.62
    - Multi: 19.81/19.63
  - Size: 840k
    - Single: 21/19.93
    - Single+DF: 21.69/20.75
    - Multi: 22.17/21.93
  - Size: 1.68m
    - Single: 23.38/23.01
    - Single+DF: 23.33/22.86
    - Multi: 23.86/23.52

- De→En
  - Size: 210k
    - Single: 11.44/11.57
    - Single+DF: 11.71/11.16
    - Multi: 12.63/12.68
  - Size: 420k
    - Single: 14.28/14.25
    - Single+DF: 14.88/15.05
    - Multi: 15.01/15.67
  - Size: 840k
    - Single: 17.09/17.44
    - Single+DF: 17.21/17.88
    - Multi: 17.33/18.14
  - Size: 1.68m
    - Single: 19.09/19.6
    - Single+DF: 19.36/20.13
    - Multi: 19.23/20.59

### Consistent improvements for large-scale translation

<table>
<thead>
<tr>
<th>Dir</th>
<th>Fr (39m)</th>
<th>Cs (12m)</th>
<th>De (4.2m)</th>
<th>Ru (2.3m)</th>
<th>Fi (2m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>→ En</td>
<td>→ En</td>
<td>→ En</td>
<td>→ En</td>
<td>→ En</td>
<td>→ En</td>
</tr>
<tr>
<td>Single</td>
<td>Single</td>
<td>Single</td>
<td>Single</td>
<td>Single</td>
<td>Single</td>
</tr>
<tr>
<td>Multi</td>
<td>Multi</td>
<td>Multi</td>
<td>Multi</td>
<td>Multi</td>
<td>Multi</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(a) BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
</tr>
<tr>
<td>Dev</td>
</tr>
<tr>
<td>Test</td>
</tr>
<tr>
<td>Test</td>
</tr>
<tr>
<td>Test</td>
</tr>
<tr>
<td>Test</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(b) LL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
</tr>
<tr>
<td>Dev</td>
</tr>
<tr>
<td>Test</td>
</tr>
<tr>
<td>Test</td>
</tr>
<tr>
<td>Test</td>
</tr>
<tr>
<td>Test</td>
</tr>
</tbody>
</table>

(Firat et al., 2016)
Multi-way, Multilingual NMT

- Consistent improvements for low-resource languages
  - the lower the training data the bigger the improvement
- In large-scale translation improves only translation to English
  - hypothesis: EN appears always as source or target language for all pairs → better decoder?

(Firat et al., 2016)
Google’s Neural Machine Translation System “Monster”

- An encoder, a decoder and an attention network
  - Plus 8-layer deep with residual connections
  - Plus refinement with Reinforcement Learning
  - Plus sub-word units...Plus....
Google’s Neural Machine Translation System “Monster”

- EN->FR training takes 6 days on 96GPUS !!!! and 3 more days for refinement...

![Table 7: Model ensemble results on WMT En→Fr (newstest2014)]

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>WPM-32K (8 models)</td>
<td>40.35</td>
</tr>
<tr>
<td>RL-refined WPM-32K (8 models)</td>
<td>41.16</td>
</tr>
<tr>
<td>LSTM (6 layers) [31]</td>
<td>35.6</td>
</tr>
<tr>
<td>LSTM (6 layers + PosUnk) [31]</td>
<td>37.5</td>
</tr>
<tr>
<td>Deep-Att + PosUnk (8 models) [45]</td>
<td>40.4</td>
</tr>
</tbody>
</table>

![Table 5: Single model results on WMT En→De (newstest2014)]

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU</th>
<th>CPU decoding time per sentence (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word</td>
<td>23.12</td>
<td>0.2972</td>
</tr>
<tr>
<td>Character (512 nodes)</td>
<td>22.62</td>
<td>0.8011</td>
</tr>
<tr>
<td>WPM-8K</td>
<td>23.50</td>
<td>0.2079</td>
</tr>
<tr>
<td>WPM-16K</td>
<td>24.36</td>
<td>0.1931</td>
</tr>
<tr>
<td>WPM-32K</td>
<td>24.61</td>
<td>0.1882</td>
</tr>
<tr>
<td>Mixed Word/Character [6]</td>
<td>24.17</td>
<td>0.3268</td>
</tr>
<tr>
<td>PBMT [37]</td>
<td>20.7</td>
<td></td>
</tr>
<tr>
<td>RNNSearch [37]</td>
<td>16.5</td>
<td></td>
</tr>
<tr>
<td>RNNSearch-LV [37]</td>
<td>16.9</td>
<td></td>
</tr>
<tr>
<td>RNNSearch-LV [37]</td>
<td>16.9</td>
<td></td>
</tr>
<tr>
<td>Deep-Att [45]</td>
<td>20.6</td>
<td></td>
</tr>
</tbody>
</table>

Data from side-by-side evaluations, where human raters compare the quality of translations for a given source sentence. Scores range from 0 to 6, with 0 meaning “completely nonsense translation”, and 6 meaning “perfect translation.”

(Wu et al., 2016)
Future of NMT and other possibilities

- **Multi-task learning**: Training multiple pairs of languages jointly and with other tasks
  → Image captioning, Speech recognition!

- **Larger context**: Modeling larger sequences than sentences as in document classification ***will be key***
  - understanding long-term dependencies
  - leveraging structural information of the input
  - being able to reason over it to solve any task
  → Effective Attention / Memory?

(Luong, Cho, Manning tutorial, 2016)
Outline of the talk

1. Recap: Word Representation Learning

2. Multilingual Word Representations
   - Alignment models
   - Evaluation tasks

3. Multilingual Word Sequence Modeling
   - Essentials: RNN, LSTM, GRU
   - Machine Translation
   - Document Classification

4. Summary

* Figure from Colah’s blog, 2015.
Paragraph vectors for Document Classification

- Learning vectors of paragraphs inspired by word2vec
  - trained without supervision on a large corpus
  - preferably similar domain as the target
- **Two methods**: with or without word ordering

(Le et al., 2014)
Paragraph vectors for Document Classification

• Learned paragraph vectors + logistic regression
• Outperformed previous method on sentence-level and document-level sentiment classification

---

**Table 1.** The performance of our method compared to other approaches on the Stanford Sentiment Treebank dataset. The error rates of other methods are reported in (Socher et al., 2013b).

<table>
<thead>
<tr>
<th>Model</th>
<th>Error rate (Positive/Negative)</th>
<th>Error rate (Fine-grained)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naïve Bayes (Socher et al., 2013b)</td>
<td>18.2 %</td>
<td>59.0 %</td>
</tr>
<tr>
<td>SVMs (Socher et al., 2013b)</td>
<td>20.6%</td>
<td>59.3%</td>
</tr>
<tr>
<td>Bigram Naïve Bayes (Socher et al., 2013b)</td>
<td>16.9%</td>
<td>58.1%</td>
</tr>
<tr>
<td>Word Vector Averaging (Socher et al., 2013b)</td>
<td>19.9%</td>
<td>67.3%</td>
</tr>
<tr>
<td>Recursive Neural Network (Socher et al., 2013b)</td>
<td>17.6%</td>
<td>56.8%</td>
</tr>
<tr>
<td>Matrix Vector-RNN (Socher et al., 2013b)</td>
<td>17.1%</td>
<td>55.6%</td>
</tr>
<tr>
<td>Recursive Neural Tensor Network (Socher et al., 2013b)</td>
<td>14.6%</td>
<td>54.3%</td>
</tr>
<tr>
<td>Paragraph Vector</td>
<td><strong>12.2%</strong></td>
<td><strong>51.3%</strong></td>
</tr>
</tbody>
</table>

**Table 2.** The performance of Paragraph Vector compared to other approaches on the IMDB dataset. The error rates of other methods are reported in (Wang & Manning, 2012).

<table>
<thead>
<tr>
<th>Model</th>
<th>Error rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>BoW (bnc) (Maas et al., 2011)</td>
<td>12.20 %</td>
</tr>
<tr>
<td>BoW (bAdv) (Maas et al., 2011)</td>
<td>11.77 %</td>
</tr>
<tr>
<td>LDA (Maas et al., 2011)</td>
<td>32.58 %</td>
</tr>
<tr>
<td>Full+BoW (Maas et al., 2011)</td>
<td>11.67 %</td>
</tr>
<tr>
<td>Full+Unlabeled+BoW (Maas et al., 2011)</td>
<td>11.11 %</td>
</tr>
<tr>
<td>WRRBM (Dahl et al., 2012)</td>
<td>12.58 %</td>
</tr>
<tr>
<td>WRRBM + BoW (bnc) (Dahl et al., 2012)</td>
<td>10.77 %</td>
</tr>
<tr>
<td>MNB-uni (Wang &amp; Manning, 2012)</td>
<td>16.45 %</td>
</tr>
<tr>
<td>MNB-bi (Wang &amp; Manning, 2012)</td>
<td>13.41 %</td>
</tr>
<tr>
<td>SVM-uni (Wang &amp; Manning, 2012)</td>
<td>13.05 %</td>
</tr>
<tr>
<td>SVM-bi (Wang &amp; Manning, 2012)</td>
<td>10.84 %</td>
</tr>
<tr>
<td>NBSVM-uni (Wang &amp; Manning, 2012)</td>
<td>11.71 %</td>
</tr>
<tr>
<td>NBSVM-bi (Wang &amp; Manning, 2012)</td>
<td>8.78 %</td>
</tr>
<tr>
<td>Paragraph Vector</td>
<td><strong>7.42%</strong></td>
</tr>
</tbody>
</table>

(Le et al., 2014)
Convolutional neural network for Document Classification

- Used multiple filter widths
- Dropout regularization (randomly dropping portion of hidden units during back-propagation)

Figure 1: Model architecture with two channels for an example sentence.

(Kim et al., 2014)
Convolutional neural network for Document Classification

<table>
<thead>
<tr>
<th>Model</th>
<th>MR</th>
<th>SST-1</th>
<th>SST-2</th>
<th>Subj</th>
<th>TREC</th>
<th>CR</th>
<th>MPQA</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN-rand</td>
<td>76.1</td>
<td>45.0</td>
<td>82.7</td>
<td>89.6</td>
<td>91.2</td>
<td>79.8</td>
<td>83.4</td>
</tr>
<tr>
<td>CNN-static</td>
<td>81.0</td>
<td>45.5</td>
<td>86.8</td>
<td>93.0</td>
<td>92.8</td>
<td>84.7</td>
<td><strong>89.6</strong></td>
</tr>
<tr>
<td>CNN-non-static</td>
<td><strong>81.5</strong></td>
<td>48.0</td>
<td>87.2</td>
<td>93.4</td>
<td>93.6</td>
<td>84.3</td>
<td>89.5</td>
</tr>
<tr>
<td>CNN-multichannel</td>
<td>81.1</td>
<td>47.4</td>
<td><strong>88.1</strong></td>
<td>93.2</td>
<td>92.2</td>
<td><strong>85.0</strong></td>
<td>89.4</td>
</tr>
<tr>
<td>RAE (Socher et al., 2011)</td>
<td>77.7</td>
<td>43.2</td>
<td>82.4</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td><strong>86.4</strong></td>
</tr>
<tr>
<td>MV-RNN (Socher et al., 2012)</td>
<td>79.0</td>
<td>44.4</td>
<td>82.9</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>RNTN (Socher et al., 2013)</td>
<td>–</td>
<td>45.7</td>
<td>85.4</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>DCNN (Kalchbrenner et al., 2014)</td>
<td>–</td>
<td>48.5</td>
<td>86.8</td>
<td>–</td>
<td>93.0</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Paragraph-Vec (Le and Mikolov, 2014)</td>
<td>–</td>
<td>–</td>
<td><strong>87.8</strong></td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>CCAE (Hermann and Blunsom, 2013)</td>
<td>77.8</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td><strong>87.2</strong></td>
</tr>
<tr>
<td>Sent-Parser (Dong et al., 2014)</td>
<td>79.5</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td><strong>86.3</strong></td>
</tr>
<tr>
<td>NBSVM (Wang and Manning, 2012)</td>
<td>79.4</td>
<td>–</td>
<td>–</td>
<td>93.2</td>
<td>–</td>
<td>81.8</td>
<td>86.3</td>
</tr>
<tr>
<td>MNB (Wang and Manning, 2012)</td>
<td>79.0</td>
<td>–</td>
<td>–</td>
<td><strong>93.6</strong></td>
<td>–</td>
<td>80.0</td>
<td>86.3</td>
</tr>
<tr>
<td>G-Dropout (Wang and Manning, 2013)</td>
<td>79.0</td>
<td>–</td>
<td>–</td>
<td>93.4</td>
<td>–</td>
<td>82.1</td>
<td>86.1</td>
</tr>
<tr>
<td>F-Dropout (Wang and Manning, 2013)</td>
<td>79.1</td>
<td>–</td>
<td>–</td>
<td><strong>93.6</strong></td>
<td>–</td>
<td>81.9</td>
<td>86.3</td>
</tr>
<tr>
<td>Tree-CRF (Nakagawa et al., 2010)</td>
<td>77.3</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>81.4</td>
<td>86.1</td>
</tr>
<tr>
<td>CRF-PR (Yang and Cardie, 2014)</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>82.7</td>
<td>–</td>
</tr>
<tr>
<td>SVM$_S$ (Silva et al., 2011)</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>95.0</td>
<td>–</td>
</tr>
</tbody>
</table>

- Not all baseline methods used drop-out though

(Kim et al., 2014)
Modeling and Summarizing Documents with a Convolutional Network

- Similar to Kim et al, 2014 however different
  - K-max pooling instead of max pooling
  - Two layers of convolutions

(Denil et al., 2014)
Modeling and Summarizing Documents with a Convolutional Network

<table>
<thead>
<tr>
<th>Model</th>
<th>Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>66</td>
</tr>
<tr>
<td>BiNB</td>
<td>62</td>
</tr>
<tr>
<td>MaxEnt</td>
<td>61</td>
</tr>
<tr>
<td>Max-TDNN</td>
<td>76</td>
</tr>
<tr>
<td>NBoW</td>
<td>68</td>
</tr>
<tr>
<td>DCNN</td>
<td>45</td>
</tr>
<tr>
<td>Our Model</td>
<td>46</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>BoW (bΔt’c)</td>
<td>88.23%</td>
</tr>
<tr>
<td>Full+BoW</td>
<td>88.33%</td>
</tr>
<tr>
<td>Full+Unlabelled+BoW</td>
<td>88.89%</td>
</tr>
<tr>
<td>WRRBM</td>
<td>87.42%</td>
</tr>
<tr>
<td>WRRBM+BoW (bnc)</td>
<td>89.23%</td>
</tr>
<tr>
<td>SVM-bi</td>
<td>86.95%</td>
</tr>
<tr>
<td>NBSVM-uni</td>
<td>88.29%</td>
</tr>
<tr>
<td>NBSVM-bi</td>
<td>91.22%</td>
</tr>
<tr>
<td>Paragraph Vector</td>
<td>92.58%</td>
</tr>
<tr>
<td>Our model</td>
<td>89.38%</td>
</tr>
</tbody>
</table>

Table 1: Left: Number of test set errors on the twitter sentiment dataset. The first block of three entries is from Go et al. [5], the second block is from Kalchbrenner et al. [13]. Right: Error rates on the IMDB movie review data set. The first block is from Maas et al. [16], the second from Dahl et al. [3], the third from Wang and Manning [24] and the fourth from Le and Mikolov [15].

(Denil et al., 2014)
Modeling and Summarizing Documents with a Convolutional Network

<table>
<thead>
<tr>
<th>Proportion</th>
<th>Summary</th>
<th>Random</th>
<th>Margin</th>
<th>Fixed</th>
<th>Summary</th>
<th>Random</th>
<th>Margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td>83.03</td>
<td>83.03</td>
<td>—</td>
<td>Pick 5</td>
<td>83.07</td>
<td>80.02</td>
<td>+3.05</td>
</tr>
<tr>
<td>50%</td>
<td>83.53</td>
<td>79.79</td>
<td>+3.74</td>
<td>Pick 4</td>
<td>83.09</td>
<td>79.05</td>
<td>+4.04</td>
</tr>
<tr>
<td>33%</td>
<td>83.10</td>
<td>76.72</td>
<td>+6.38</td>
<td>Pick 3</td>
<td>82.88</td>
<td>77.15</td>
<td>+5.73</td>
</tr>
<tr>
<td>25%</td>
<td>82.91</td>
<td>74.87</td>
<td>+8.04</td>
<td>Pick 2</td>
<td>82.04</td>
<td>74.48</td>
<td>+7.56</td>
</tr>
<tr>
<td>20%</td>
<td>82.67</td>
<td>73.20</td>
<td>+9.47</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First and last</td>
<td>68.62</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Results of classifying summaries with Naïve Bayes. Results labelled proportion indicate selecting up to the indicated percentage of sentences in the review, and results labelled fixed show the result of selecting a fixed number of sentences from each. The summary column shows the accuracy of Naïve Bayes on summaries produced by our model. The random column shows the same model classifying summaries created by selecting sentences at random. The margin column shows the difference in accuracy between our model and the random summaries.

(Denil et al., 2014)
Modeling and Summarizing Documents with a Convolutional Network

Figure 3: Several example summaries created by our ConvNet. The full text of the review is shown in black and the sentences selected by the ConvNet appear in colour. While summarising a review with the first sentence is a popular pragmatic approach, it is clear in these examples that this heuristic is not as effective as the ConvNet summarisation scheme. Each summary is created by selecting up to 20% of the sentences in the review.

(Denil et al., 2014)
Gated recurrent neural network for Document Classification

(Tang et al., 2015)
## Gated recurrent neural network for Document Classification

<table>
<thead>
<tr>
<th>Method</th>
<th>Yelp 2013</th>
<th>Yelp 2014</th>
<th>Yelp 2015</th>
<th>IMDB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accuracy</td>
<td>MSE</td>
<td>Accuracy</td>
<td>MSE</td>
</tr>
<tr>
<td>Majority</td>
<td>0.356</td>
<td>3.06</td>
<td>0.361</td>
<td>3.28</td>
</tr>
<tr>
<td>SVM + Unigrams</td>
<td>0.589</td>
<td>0.79</td>
<td>0.600</td>
<td>0.78</td>
</tr>
<tr>
<td>SVM + Bigrams</td>
<td>0.576</td>
<td>0.75</td>
<td>0.616</td>
<td>0.65</td>
</tr>
<tr>
<td>SVM + TextFeatures</td>
<td>0.598</td>
<td>0.68</td>
<td>0.618</td>
<td>0.63</td>
</tr>
<tr>
<td>SVM + AverageSG</td>
<td>0.543</td>
<td>1.11</td>
<td>0.557</td>
<td>1.08</td>
</tr>
<tr>
<td>SVM + SSWE</td>
<td>0.535</td>
<td>1.12</td>
<td>0.543</td>
<td>1.13</td>
</tr>
<tr>
<td>JMARS</td>
<td>N/A</td>
<td>–</td>
<td>N/A</td>
<td>–</td>
</tr>
<tr>
<td>Paragraph Vector</td>
<td>0.577</td>
<td>0.86</td>
<td>0.592</td>
<td>0.70</td>
</tr>
<tr>
<td>Convolutional NN</td>
<td>0.597</td>
<td>0.76</td>
<td>0.610</td>
<td>0.68</td>
</tr>
<tr>
<td>Conv-GRNN</td>
<td>0.637</td>
<td>0.56</td>
<td>0.655</td>
<td>0.51</td>
</tr>
<tr>
<td>LSTM-GRNN</td>
<td><strong>0.651</strong></td>
<td><strong>0.50</strong></td>
<td><strong>0.671</strong></td>
<td><strong>0.48</strong></td>
</tr>
</tbody>
</table>

Table 2: Sentiment classification on Yelp 2013/2014/2015 and IMDB datasets. Evaluation metrics are accuracy (higher is better) and MSE (lower is better). The best method in each setting is in **bold**. (Tang et al., 2015)
Standard Pipeline for Document Classification

- **Feature engineering**: BOW, n-grams, topic models, etc.
- **Feature learning**: auto-encoders, convolutional, recurrent, recursive NNs

Limitations
- Treat the text globally and ignore the weak nature of labels
- Make simplistic assumptions when aggregating or pooling features
- Offer few means for model interpretation

(Pappas and Popescu-Belis, 2014)
Multiple-instance Learning for Document Classification

Given $\mathcal{D} = \{(b_{ij}, y_i) \mid j = 1 \ldots n_i\}^m$, find $\Phi_k : \mathcal{B} \rightarrow \mathcal{X} \rightarrow \mathcal{Y}_k$

- The bag $B_i$ is a review represented by $n_i$ instances $b_{ij}$, its sentences
- The labels $y_i \in \mathbb{R}^k$ are the aspect ratings of the review
- The exemplar (representation) $x_i \in \mathbb{R}^d$ of $B_i$ is initially unknown

Advantages

- Several input assumptions (Aggregated, Instance, Prime, Clustering)
- Subsumes traditional supervised regression (Aggregated)
- Better suited for weak labels, interpretable and flexible

(Pappas and Popescu-Belis, 2014)
How to combine vectors?

Structural assumptions

1. **Aggregated instances**: sum or average instances
   \[
   f \leftarrow D_{agg} = \{(x_i, y_i) \mid i = 1, \ldots, m\}
   \]
   \[
   \hat{y}(B_i) = f(x_i) = f(\text{mean}(\{b_{ij} \mid w_j = 1, \ldots, n_i\}))
   \]

2. **Instance-as-example**: instances inherit bag labels
   \[
   f \leftarrow D_{ins} = \{(b_{ij}, y_i) \mid j = 1, \ldots, n_i; i = 1, \ldots, m\}
   \]
   \[
   \hat{y}(B_i) = \text{mean}(\{f(b_{ij}) \mid j = 1, \ldots, n_i\})
   \]

3. **Prime instance**: a single instance is selected
   \[
   f \leftarrow D_{pri} = \{(b_i^p, y_i) \mid i = 1, \ldots, m\}
   \]
   \[
   \hat{y}(B_i) = \text{mean}(\{f(b_{ij}) \mid j = 1, \ldots, n_i\})
   \]

(Pappas and Popescu-Belis, 2014)
Joint learning of an instance relevance mechanism and a classifier

Inspired from method proposed by Wagstaff and Lane (2007):

\[ x_i = \sum_{j=1}^{n_i} \psi_{ij} b_{ij}, \quad \psi_{ij} \geq 0 \quad \text{and} \quad \sum_{j=1}^{n_i} \psi_{ij} = 1 \]

1. Models both instance weights and target labels
   - Target labels model: \( \hat{y}_i = f(\Phi, B_i) = \Phi^T(B_i\psi_i) \)
   - Instance weights model: \( \hat{\psi}_i = f(O, B_i)O^TB_i \)
2. Defines loss based on regularized least squares
   - Supports large datasets and high dimensionality \( \mathcal{O}(md^2) \)
   - Adapts to domain data through regularization

(Pappas and Popescu-Belis, 2014)
Joint differentiable objective for solving with SGD

Based on stochastic gradient descent

\[ \sigma(B_i, O) = P(\psi = y_i | B_i) = \frac{\exp(O^T B_i)}{\sum_{k=1}^{n_i} \exp(O^T B_{ik})} \]

\[ O, \Phi = \arg \min_{O, \Phi} \sum_{i=1}^{\ldots} (y_i - \Phi^T (B_i \cdot \sigma(B_i, O)))^2 + \Omega(\Phi, O) \]

- Preserves constraints of instance relevance assumption
- Achieves similar performance to alternating projections
- Makes the learning procedure more scalable

Shared material
→ Code: wmil, wmil-sgd

https://github.com/nik0spapp/

(Pappas and Popescu-Belis, 2014)
Observations on aspect rating prediction

- The proposed mechanism is superior than alternatives
  - all text regions are useful but to a different extent
- Benefit regardless of the input features used
- Reaches state-of-the-art without using:
  - structured output learning
  - segmented text

(Pappas and Popescu-Belis, 2014)
Comparison with neural network models

- This mechanism can be used as a parametric pooling function of NNs
  - operating on intermediate hidden states
- Works better than Dense, GRU neural methods + average pooling
- Outperforms RCNN and uses far less parameters

Table 2: Comparison of our instance relevance mechanism (MIR) integrated within neural networks, with state-of-the-art neural networks, on the aspect rating prediction task in terms of mean squared error (MSE). $|\theta|$ indicates the number of parameters.

(Pappas and Popescu-Belis, 2016)
Hierarchical attention networks for Document Classification

- Very similar hierarchical structure as Tang et al., 2015 except average pooling
- attention mechanism at the word and document levels

\[
\begin{align*}
\alpha_{it} &= \frac{\exp(u_{it}^T u_w)}{\sum_t \exp(u_{it}^T u_w)} \\
\alpha_{it} h_{it} &= \sum_t \alpha_{it} h_{it} \\
u_{it} &= \tanh(W_w h_{it} + b_w)
\end{align*}
\]

(Yang et al., 2016)
Hierarchical attention networks for Document Classification

<table>
<thead>
<tr>
<th>Methods</th>
<th>Yelp’13</th>
<th>Yelp’14</th>
<th>Yelp’15</th>
<th>IMDB</th>
<th>Yahoo Answer</th>
<th>Amazon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zhang et al., 2015</td>
<td>BoW</td>
<td>-</td>
<td>58.0</td>
<td>-</td>
<td>68.9</td>
<td>54.4</td>
</tr>
<tr>
<td></td>
<td>BoW TFIDF</td>
<td>-</td>
<td>59.9</td>
<td>-</td>
<td>71.0</td>
<td>55.3</td>
</tr>
<tr>
<td></td>
<td>ngrams</td>
<td>-</td>
<td>56.3</td>
<td>-</td>
<td>68.5</td>
<td>54.3</td>
</tr>
<tr>
<td></td>
<td>ngrams TFIDF</td>
<td>-</td>
<td>54.8</td>
<td>-</td>
<td>68.5</td>
<td>52.4</td>
</tr>
<tr>
<td></td>
<td>Bag-of-means</td>
<td>-</td>
<td>52.5</td>
<td>-</td>
<td>60.5</td>
<td>44.1</td>
</tr>
<tr>
<td>Tang et al., 2015</td>
<td>Majority</td>
<td>35.6</td>
<td>36.1</td>
<td>36.9</td>
<td>17.9</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVM + Unigrams</td>
<td>58.9</td>
<td>60.0</td>
<td>61.1</td>
<td>39.9</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVM + Bigrams</td>
<td>57.6</td>
<td>61.6</td>
<td>62.4</td>
<td>40.9</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVM + TextFeatures</td>
<td>59.8</td>
<td>61.8</td>
<td>62.4</td>
<td>40.5</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVM + AverageSG</td>
<td>54.3</td>
<td>55.7</td>
<td>56.8</td>
<td>31.9</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVM + SSWE</td>
<td>53.5</td>
<td>54.3</td>
<td>55.4</td>
<td>26.2</td>
<td>-</td>
</tr>
<tr>
<td>Zhang et al., 2015</td>
<td>LSTM</td>
<td>-</td>
<td>58.2</td>
<td>-</td>
<td>70.8</td>
<td>59.4</td>
</tr>
<tr>
<td></td>
<td>CNN-char</td>
<td>-</td>
<td>62.0</td>
<td>-</td>
<td>71.2</td>
<td>59.6</td>
</tr>
<tr>
<td></td>
<td>CNN-word</td>
<td>-</td>
<td>60.5</td>
<td>-</td>
<td>71.2</td>
<td>57.6</td>
</tr>
<tr>
<td>Tang et al., 2015</td>
<td>Paragraph Vector</td>
<td>57.7</td>
<td>59.2</td>
<td>60.5</td>
<td>34.1</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>CNN-word</td>
<td>59.7</td>
<td>61.0</td>
<td>61.5</td>
<td>37.6</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Conv-GRNN</td>
<td>63.7</td>
<td>65.5</td>
<td>66.0</td>
<td>42.5</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>LSTM-GRNN</td>
<td>65.1</td>
<td>67.1</td>
<td>67.6</td>
<td>45.3</td>
<td>-</td>
</tr>
<tr>
<td>This paper</td>
<td>HN-AVE</td>
<td>67.0</td>
<td>69.3</td>
<td>69.9</td>
<td>47.8</td>
<td>75.2</td>
</tr>
<tr>
<td></td>
<td>HN-MAX</td>
<td>66.9</td>
<td>69.3</td>
<td>70.1</td>
<td>48.2</td>
<td>75.2</td>
</tr>
<tr>
<td></td>
<td>HN-ATT</td>
<td>68.2</td>
<td>70.5</td>
<td>71.0</td>
<td>49.4</td>
<td>75.8</td>
</tr>
</tbody>
</table>

Table 2: Document Classification, in percentage

(Yang et al., 2016)
Reflections on Multilingual Document Classification

- **What are the present limitations?**
  - Current evaluation datasets contain small number of target classes and examples
    - RCV1/RCV2 $\rightarrow$ 6,000 documents, 2 langs, 4 labels
    - TED corpus $\rightarrow$ 12,078 documents, 12 langs, 15 labels
  - Requires the labels to be common across languages
  - Data are not enough to train SOA neural architectures

- **Observation:** currently there are several domains which support multiple languages but only monolingual classification is possible
New dataset: Deutsche Welle corpus (600k docs, 8 langs)

Table 1: Deutsche Welle corpus statistics.

<table>
<thead>
<tr>
<th>Language</th>
<th>Documents</th>
<th>$Y_g$</th>
<th>$Y_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>112,816</td>
<td>327</td>
<td>1,058</td>
</tr>
<tr>
<td>German</td>
<td>132,709</td>
<td>367</td>
<td>809</td>
</tr>
<tr>
<td>Spanish</td>
<td>75,827</td>
<td>159</td>
<td>684</td>
</tr>
<tr>
<td>Portuguese</td>
<td>39,474</td>
<td>95</td>
<td>301</td>
</tr>
<tr>
<td>Ukrainian</td>
<td>35,423</td>
<td>28</td>
<td>260</td>
</tr>
<tr>
<td>Russian</td>
<td>108,076</td>
<td>102</td>
<td>814</td>
</tr>
<tr>
<td>Arabic</td>
<td>57,697</td>
<td>91</td>
<td>344</td>
</tr>
<tr>
<td>Persian</td>
<td>36,282</td>
<td>71</td>
<td>127</td>
</tr>
</tbody>
</table>
Conclusion

• Multilingual word embeddings are useful for tasks where there is lack of parallel data
• Word sequence modeling is advancing quickly with the establishment of neural methods
  • Machine Translation
  • Document Classification
• **Multilingual Neural Machine Translation**
  • is useful for low-resourced languages
  • transfers knowledge in large-scale setting
• **Multilingual Document Classification**
  • several large resources available but with disjoint labels
  • could possibly benefit from NMT lessons
References (1/3)


References (2/3)

References (3/3)

• Faruqui, Manaal, Jesse Dodge, Sujay K. Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. "Retrofitting word vectors to
Resources (1/2)

‣ **Online courses**
  • Coursera course on “Neural networks for machine learning” by Geoffrey Hinton
  • Coursera course on “Machine learning” by Andrew Ng
    [https://www.coursera.org/learn/machine-learning](https://www.coursera.org/learn/machine-learning)
  • Stanford CS224d “Deep learning for NLP” by Richard Socher
    [http://cs224d.stanford.edu/](http://cs224d.stanford.edu/)

‣ **Conference tutorials**
    [http://www.emnlp2015.org/tutorials.html#t1](http://www.emnlp2015.org/tutorials.html#t1)
  • Mitesh M Kharpa, Sarath Chandar, “Multilingual and Multimodal Language Processing”, NAACL 2016 tutorial.
Resources (2/2)

→ Deep learning toolkits
  • Theano http://deeplearning.net/software/theano
  • Torch http://www.torch.ch/
  • Tensorflow http://www.tensorflow.org/
  • Keras http://keras.io/

→ Pre-trained word vectors and codes
  • Word2vec toolkit and vectors
    https://code.google.com/p/word2vec/
  • GloVe code and vectors
    http://nlp.stanford.edu/projects/glove/
  • Hellinger PCA
    https://github.com/rlebret/hpca
  • Online word vector evaluation
    http://wordvectors.org/