Learning text representations from character-level data

Grzegorz Chrupała

Department of Communication and Information Sciences
Tilburg University

CLIN 2013

Text representations

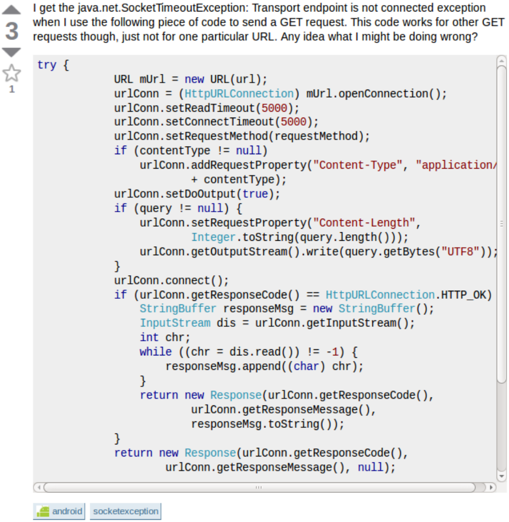
- Traditionally focused on **word** level
  - Brown or HMM word classes
  - Collobert and Weston distributed representations
  - LDA-type soft classes
- Successfully used as features in
  - Chunking and named entity recognition
  - Parsing
  - Semantic relation labeling

Limitations

Assuming words as input not always realistic

- Agglutinative and other morphologically complex languages
- Naturally occurring text: often mix NL strings comingled with other character data

Sample post on STACKOVERFLOW



I get the java.net.SocketTimeoutException: Transport endpoint is not connected exception when I use the following piece of code to send a GET request. This code works for other GET requests though, just not for one particular URL. Any idea what I might be doing wrong?

```
try {
            URL mUrl = new URL(url);
            urlConn = (HttpURLConnection) mUrl.openConnection();
            urlConn.setReadTimeout(5000);
            urlConn.setConnectTimeout(5000);
            urlConn.setRequestMethod(requestMethod);
            if (contentType != null)
                urlConn.addRequestProperty("Content-Type", "application/
                        + contentType);
            urlConn.setDoOutput(true);
            if (query != null) {
                urlConn.setRequestProperty("Content-Length",
                        Integer.toString(query.length()));
                urlConn.getOutputStream().write(query.getBytes("UTF8"));
            }
            urlConn.connect();
            if (urlConn.getResponseCode() == HttpURLConnection.HTTP_OK)
                StringBuffer responseMsg = new StringBuffer();
                InputStream dis = urlConn.getInputStream();
                int chr;
                while ((chr = dis.read()) != -1) {
                    responseMsg.append((char) chr);
                }
                return new Response(urlConn.getResponseCode(),
                        urlConn.getResponseMessage(),
                        responseMsg.toString());
            }
            return new Response(urlConn.getResponseCode(),
                    urlConn.getResponseMessage(), null);
```

Segmentation of the character stream

- To define tokenization meaningfully
- First need to segment and label character data
  - English
  - Code block (Java, Python...)
  - Inline code
  - ...

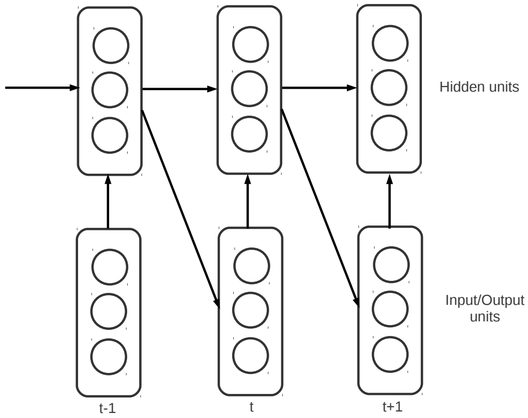
Test case for inducing text representation

- STACKOVERFLOW HTML markup as supervision signal
- Character-level sequence model (CRF)
- Character n-gram features as baseline

→ Add text representation features

→ Learned from raw character data (no labels)

Simple Recurrent Neural Network (Elman net)



- Current input and previous state combined to create current state
- Output is generated by current state
- **Self-supervised**

Hidden units

Hidden units

- Encode history
- Hopefully, generalize

Sample of nearest neighbors according to cosine of the hidden layer activation in a span of 10.000 characters

writing·a·.NET·applicati ·any·links·with·informati d·to·test·a·IP·verificati enerate·each·IP·combinati ·files.·I·have·presentati	p":·{"last_share":·130738 c":·{"last_share":·130744 p":·{"last_share":·130744 :·{"last_share":·13073896 :·{"last_share":·13074418
o·$n1.’.’.$n2.’.’.$n3.’.’ $n1.’.’.$n2.’.’.$n3++.’.’ t;’;¶········echo·$n1.’.’ ·····echo·$n1.’.’.$n2.’.’ ·····echo·$n1.’.’.$n2.’.’	able·has·integer·values·a 5.·For·all·these·values·I lots·of·private·methods·a me·across·any·resources·e an·add·more·connections·s

Generated random text

```
I·only·make·event·glds.

so,·on·the·cell·proceedclicks·like·completed,·with·color?

····st·potention,
‘column’]HeaderException=ID·=·new·Put="True"·MetadataTemplate,
·grwTrowerRow="SELECTEMBRow"·on?

All·clearBeanLockCollection="#7293df3335b-E9"·/&gt;
············&lt;Image:DataKey="BackgroundCollectionC2UTID"·
onclick="Nore"·
```

Segmentation and labeling of Stackoverflow posts

- Generate labels from HTML markup
- From trained RNN model
  - Run on labeled train and test data
  - Record hidden unit activations at each position in text
  - Use as extra features for CRF

Labels

Block

w	r	o	n	g	?	¶	t	r	y
O	O	O	O	O	O	O	B-BL	I-BL	I-BL

Inline

e	r	·	.	.	/	i	m	g
O	O	O	B-IN	I-IN	I-IN	I-IN	I-IN	I-IN

Baseline feature set

`...wrong?¶try {...`

Unigram	n g ? ¶ t
Bigram	g? ?¶
Trigram	g?¶
Fourgram	ng?¶ g?¶t
Fivegram	ng?¶t

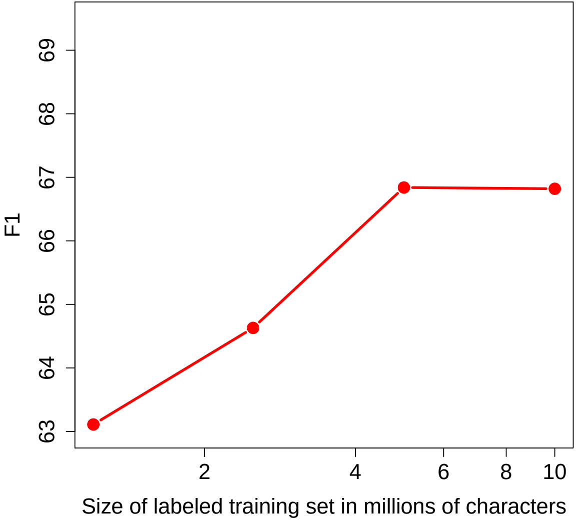
Augmented feature set

- Baseline features
- 400-unit hidden layer activation
  - For each of 10 most active units
    - Is the activation > 0.5?

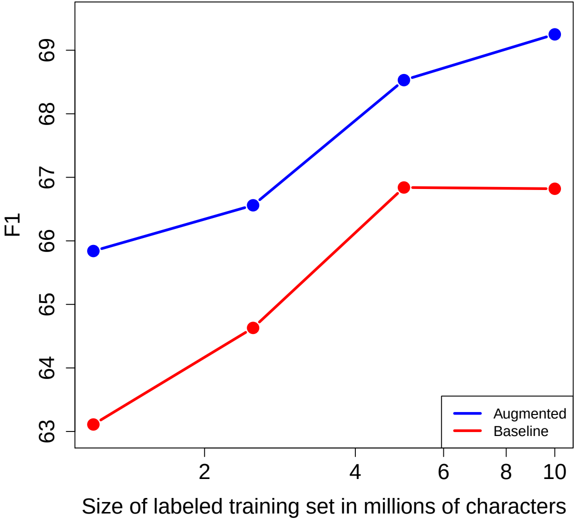
Data sets

- Labeled
  - Train: 1.2 – 10 million characters
  - Test: 2 million characters
- Unlabeled
  - 465 million characters

Baseline F-score



Augmented


F1
63
64
65
66
67
68
69
2
4
6
8
10
Size of labeled training set in millions of characters
Augmented
Baseline

Details (best model)

Label	Precision	Recall	F-1
All	83.6	59.1	69.2
BLOCK	90.8	90.6	90.7
INLINE	40.8	10.5	16.7

- Sequence accuracy: 70.7%
- Character accuracy: 95.2%

Conclusion

Simple Recurrent Networks learn abstract distributed representations useful for character level prediction tasks.

Future work

- Alternative network architecture: Sutskever et al. 2011, dropout
- Distributed analog of bag-of-words
- Test on other tasks/datasets