Learning text embeddings with recurrent neural language models

Grzegorz Chrupała

Tilburg University

Delft 2014

Linguistic data

- Speech: acoustic signal
- Text: streams of characters/bytes
- In order to understand them, add many layers of annotation.

Linguistic analyses

A greet recipe 4 pulpo a la gallega!
[EN] [ES]
A **great** recipe **for** pulpo a la gallega!
D A Noun P Noun P D Noun !
[A [great recipe] [for [pulpo a la gallega]]]

- Most analyses are variations of sequence labeling
- Deeper analyses often build tree or graph structures
- **This talk: sequence labeling**

Traditional NLP

- Supervised learning
- Linear models
- Complex manually engineered features

Some recent developments

1. Language models based on recurrent neural networks (Mikolov)
2. Word embeddings induced from unlabeled data (Collobert & Weston, ...)
3. Recursive autoencoders for composing word- into sentence representations (Socher)

Character-level text representations

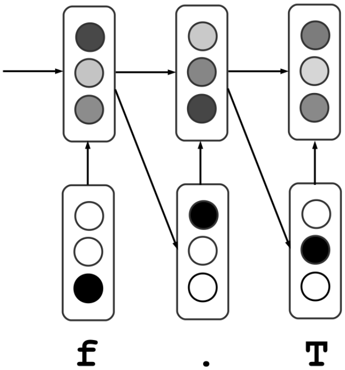
This research brings together:

- Linear models for sequences (Conditional Random Fields)
- Simple Recurrent Network (SRN) for language modeling
- Representations/embeddings as features with a twist: at character/byte level

Architecture

- Inputs
  - Large amount of raw text
  - Smallish amount of labeled text
- Process
  - Build feature extractor by training SRN on raw text (unsupervised)
  - Build sequence labeler by training CRF on labeled data extended with extracted features (supervised)

Simple recurrent networks



Elman, J. L. (1990). Finding structure in time.

Character-level text representations

- Trained to predict next character in sequence
- Hidden layer stores a compressed representation of seen characters
- Encodes generalizations
- Embeds string at each position in a low-dimensional space

Visualizing embeddings

SRN trained on 400.000 bytes of Twitter stream. Some nearest neighbors in 400-dimensional embedding space.

should h	should d	will s	will m	should a
@justth	@neenu	@raven_	@lanae	@despic
maybe	u maybe y	cause i	wen i	when i

Tweets randomly generated from the trained SRN language model

```
@YuszLAL100A 暇すぎるｗｗｗｗｗとか麺役者についてる…（ゝ >
晒せ 信じに行けていいんだな…。 RT @yaepdrrafa:
@fsch_chany siaaa,, dobek taha subus sama kiri kabur
wanak... hahah
なかなかない。
やばい
But I'm the good first-Good Chulc
```

Tasks

- Detect code blocks embedded in natural language text
- Segment text into words and sentences
- **Normalize tweets**

Chrupała, G. (2014). Normalizing tweets with edit scripts and recurrent neural embeddings. ACL.

Tweets and similar user-generated content

- Heterogeneous in style (from slangy to formal)
- Frequent mis- and respellings and abbreviations
- Non-standard vocabulary
- Non-standard syntax
- ...

Normalization

Convert text to a canonical, normalized form

- Expand abbreviations
- Correct spellings
- Replace non-standard words

In hope of making text easier to process/understand for downstream applications

Normalization examples

I will c wat i can do I will see what I can do
imma jus start puttn it out there I'm going to just start putting it out there

Noisy-channel model

$$\text{P(target}|\text{original)} \propto$$

$$\underbrace{\text{P(original}|\text{target)}}_{\text{Noise model}} \times \underbrace{\text{P(target)}}_{\text{Language model}}$$

- Noise model: which respellings are probable? E.g. dictionaries
- Language model: which target strings are probable? Trained on large amounts of target data.

Noisy-channel approach

- Good match for spell-checking formal text
  - Large amounts or proof-read formal newspaper text to train language model on
- **What kind of text to use for tweet normalization?**
  - **Newspaper text?**
  - **Transcribed spoken language?**

Direct approach

$$\hat{\text{target}} = \operatorname{argmax}_{\text{target}} \text{P}(\text{diff}(\text{original}, \text{target})|\text{original})$$

- P is a linear-chain Conditional Random Field model
- diff(original, target) is a series of string edits which transform original to target

- P is trained on labeled examples (original-target pairs)
- No explicit target language model
- Instead, bring in information from unlabeled **original data** via features learned with SRN LM on lots of unedited tweets

Diff: Edit script

Input	c	␣	w	a	t
Diff	DEL	INS(see)	NIL	INS(h)	NIL
Output		see␣	w	ha	t

- Each position in input string associated with edit operation.
  - A sequence labeling task

Features

- Baseline features: byte n-grams

  `c _ w a t c_ _w wa at c_w _wa wat c_wa _wat c_wat`
- SRN features
  - SRN trained on 400 MB of raw Twitter feed.
  - Activations of 400 hidden units when network is predicting current byte.
  - Discretized: for 10 most active units, on/off with threshold 0.5.

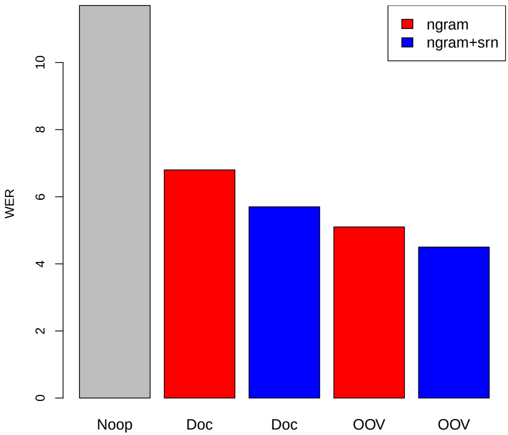
Dataset

- Tweet normalization dataset from Han and Baldwin 2011
- 549 tweets, with normalized versions
- Only word-to-word transformations

Model versions

- No-op: make no changes
- Doc: train on and label whole tweets
- OOV: train on and label OOV-words

Word error rates



Compared to previous work

Method	WER (%)
No-op	11.2
HB-dict	6.6
GHM-dict	7.6
S-dict	9.7
Dict-combo	4.9
Dict-combo+HB-norm	7.9
oov-only ngram+srn (test)	**4.8**

Where SRN features help

9 cont continued	5 gon gonna
4 bro brother	4 congrats congratulations
3 yall you	3 pic picture
2 wuz what's	2 mins minutes
2 juss just	2 fb facebook

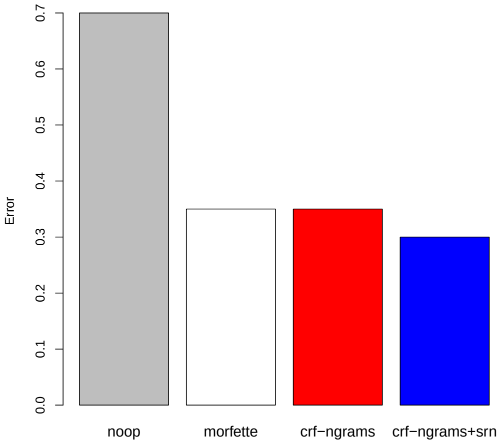
Lemmatization of historical text

(Work in progress with Mike Kestemont)

Wanneer	nu	die	ynnige	ziel
wanneer	nu	de	innig	ziel

hoer	selven	in	god	dus	ontsoncken ...
zich	zelf	in	god	dus	ontzinken ...

Preliminary results

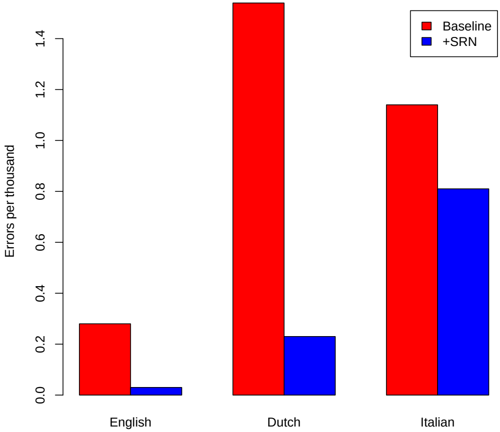


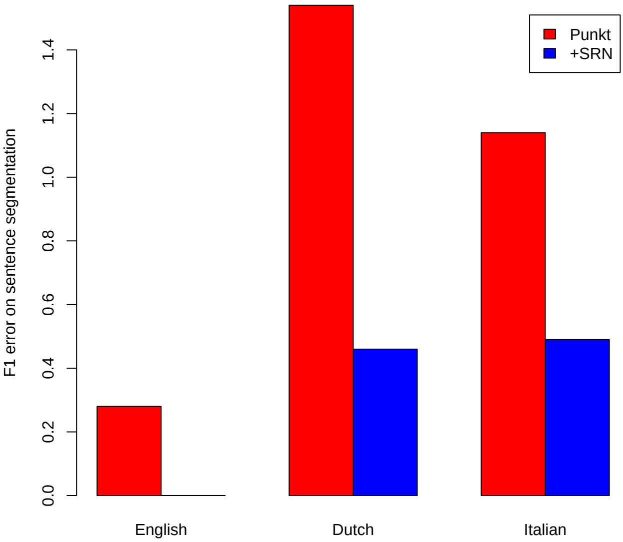
Word and sentence segmentation

Evang, K., Basile, V., Chrupała, G., & Bos, J. (2013). Elephant: Sequence Labeling for Word and Sentence Segmentation. EMNLP

```
It didn't matter if the faces were male,
S-.T--T--.T-----.T-.T--.T----.T---.T---T.
female or those of children. Eighty-
T-----.T-.T----.T-.T-------T.S------.
three percent of people in the 30-to-34
-----.T------.T-.T-----.T-.T--.T-------.
year old age range gave correct responses.
T---.T--.T--.T----.T---.T------.T--------T
```

Results


Baseline
+SRN
Errors per thousand
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
English
Dutch
Italian

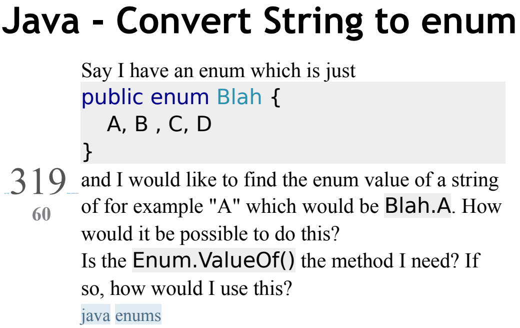

F1 error on sentence segmentation
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
Punkt
+SRN
English
Dutch
Italian

Where SRN features helped

	`prof. Teulings het`
	`T---T.S-------.T--.`
+SRN	`T----.T-------.T--.`
	` bleek 0,4 procent`
	`.T----.TT-.T------.`
+SRN	`.T----.T--.T------.`
	`per costringerlo al`
	`T--.T-----------.T-`
+SRN	`T--.T---------T-.T-`

Labeling code blocks

Chrupała, G. (2013). Text segmentation with character-level text embeddings. Deep Learning for Audio, Speech and Language Processing (ICML)



Java - Convert String to enum

319

60

Say I have an enum which is just

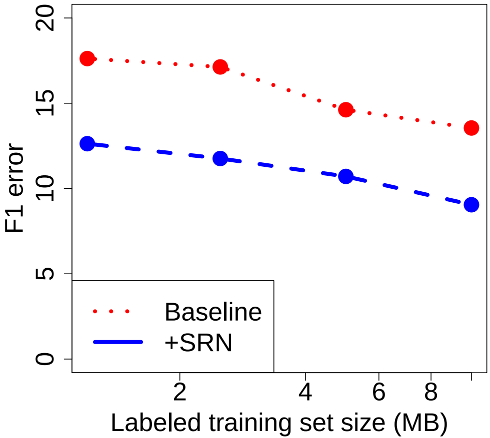
```
public enum Blah {
    A, B , C, D
}
```

and I would like to find the enum value of a string of for example "A" which would be `Blah.A`. How would it be possible to do this?

Is the `Enum.ValueOf()` the method I need? If so, how would I use this?

java enums

Labeled examples equivalence



Conclusion

Features learned by SRNs when combined with linear sequence models:

- Improve performance
- Or reduce amount of supervision needed

Future

- Work in progress on applications
  - Lemmatization of historical language
  - Code-switching in tweets
- Feature extraction vs integrated recurrent network models

Thank you

Missed transformations

4 1 one	2 withh with
2 uu you	2 tonite tonight
2 thx thanks	2 thiis this
2 smh somehow	2 outta out
2 n in	2 m am
2 hmwrk homework	2 gf girlfriend
2 fxckin fucking	2 dha the
2 de the	2 d the
2 bhee be	2 bb baby

Sizes of datasets

Dataset	Lang	Labeled	Unlabeled
Stack	en	10.00M	465.0M
Tweet	en	0.02M	414.0M
Elephant	en	0.32M	2.5M
Elephant	nl	4.30M	43.0M
Elephant	it	4.30M	39.0M