Learning to normalize tweets with few examples

Grzegorz Chrupała | Tilburg University

Normalizing tweets

Convert tweets to canonical form easy to understand for downstream applications

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

Approaches

- Noisy-channel-style
- Finite-state transducers
- Dictionary-based
  - Hand-crafted
  - Automatically constructed

Labeled vs unlabeled data

- Noisy-channel:
  $P(\text{target}|\text{source}) \propto P(\text{source}|\text{target}) \times P(\text{target})$
  labeled unlabeled

- Dictionary lookup:
  - Induce dictionary from unlabeled data
  - Labeled data for parameter tuning

Discriminative model

$$\operatorname{argmax}_{\text{target}} \mathbf{P}(\mathbf{diff}(\text{source}, \text{target}) \mid \text{source})$$

- $\mathbf{diff}(\cdot,\cdot)$ transforms source to target
- $\mathbf{P}(\cdot)$ is some sequence model, e.g.
  - Conditional Random Fields
  - Structured Perceptron

We can include additional sources of information via features

- features derived from dictionaries
- features derived from raw text

diff - Edit script

Input	c	_	w	a	t
diff	DEL	INS(see)	NIL	INS(h)	NIL
Output		see_	w	ha	t

Each position in string labeled with edit op

Features

- Byte n-grams

  `c _ w a t c_ _w wa at c_w _wa`

  `wat c_wa _wat c_wat`

- Features derived from external resources
  - word classes
  - text representation
  - dictionary

Soft word classes

- words represented as distributions over classes
- trained with Latent Dirichlet Allocation

Chrupała (2011). Efficient induction of probabilistic word classes with LDA. IJCNLP
bitbucket.org/gchrupala/lda-wordclass

Byte-level neural text embeddings

- each position in a string represented as a 400-dimensional vector
- trained using a recurrent neural LM

Chrupała, (2013). Text segmentation with character-level text embeddings. DLASLP
Chrupała (2014). Normalizing tweets with edit scripts and recurrent neural embeddings. ACL

Dictionary of internet slang (noslang.com)

tix	tickets	2nite	tonight
tks	thanks	2nyt	tonight
tld	told	2sday	tuesday
tlk	talk	2tali	totally
tlkin	talking	304	hoe
tlkn	talking	31337	elite
tmmrw	tomorrow	4eva	forever

Dictionary

- Generate diffs between source and target of each entry
- Use diffs as a features

Example

Byte	N-grams	Word class	Text rep	Dictionary	Label
c	c_	1 1 0 0	0 0 1 0	?	DEL
_	c_	0 0 0 0	0 0 0 0	?	INS(see)
w	wa wat	0 1 0 1	1 1 0 1	NIL	NIL
a	wa at wat	0 1 0 1	1 0 0 1	INS(h)	INS(h)
t	at wat	0 1 0 1	0 0 0 1	NIL	NIL

Dataset

- Han, B., & Baldwin, T. (2011). Lexical normalisation of short text messages: Makn sens a# twitter. In *ACL*.

- 549 tweets, with normalized versions
- Only lexical normalizations

Model variant

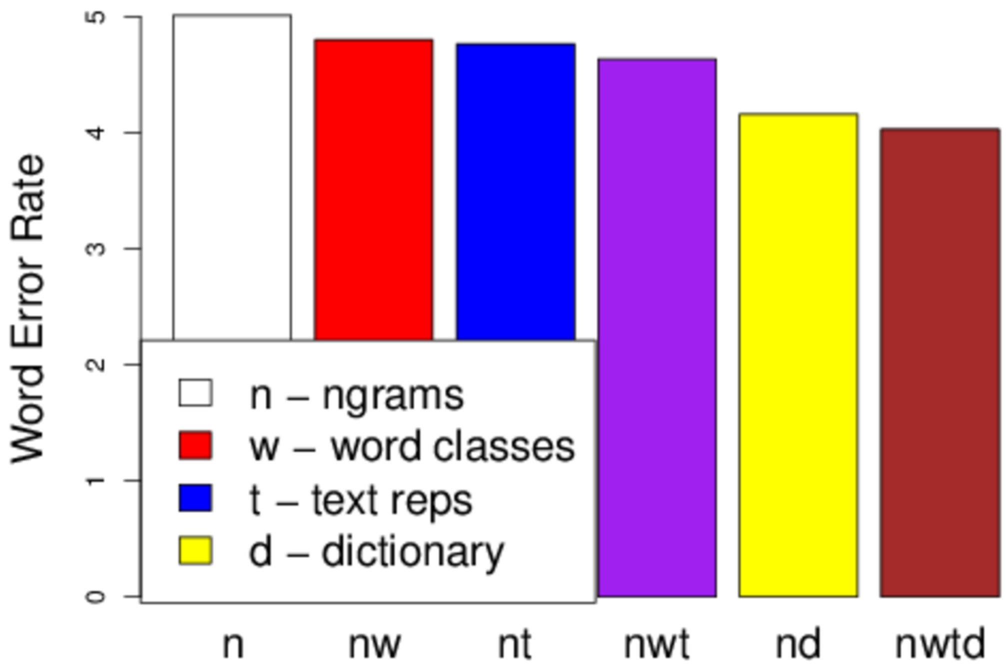
- Structured Perceptron

  bitbucket.org/gchrupala/sequor

- Word-by-word

- Only OOV words are changed

Results



cross-validation on five
development folds

Compared to Han & Bo 2012

Method	WER (%)
No-op	11.2
S-dict	9.7
GHM-dict	7.6
HB-dict	6.6
Dict-combo	4.9
nwtd	**4.0**

Where extra features helped

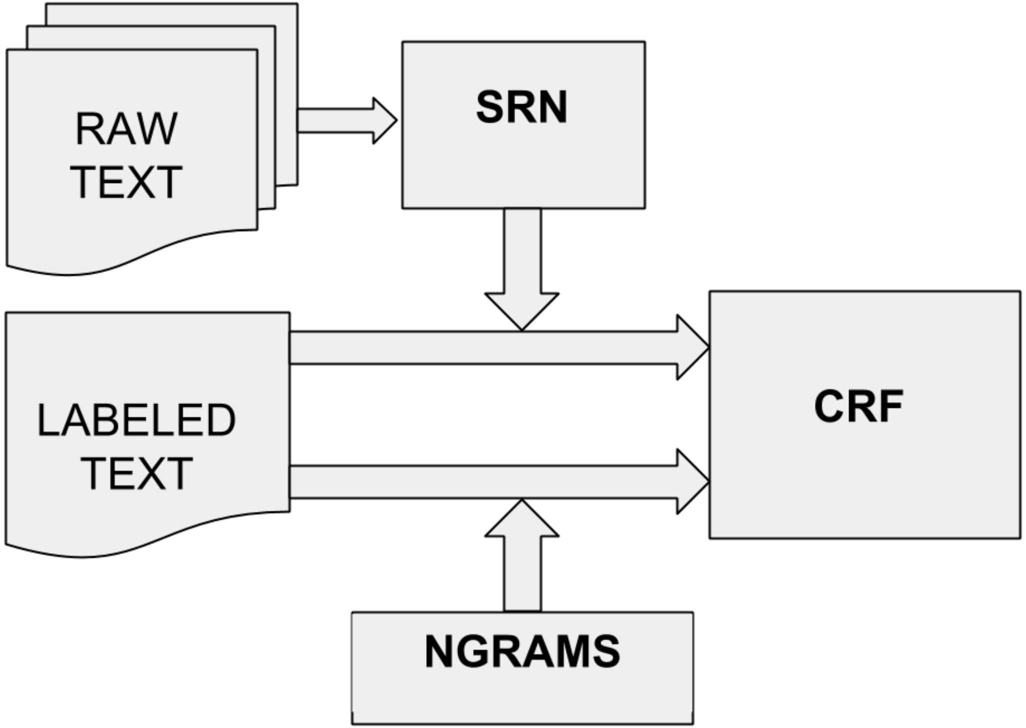
5	cont continued	5	2 to
4	gon gonna	3	congrats congratulations
2	pic picture	2	mins minutes
2	m am	1	yesss yes
1	whateva whatever	1	wasss was
1	nvr never	1	sumthings somethings

Conclusion

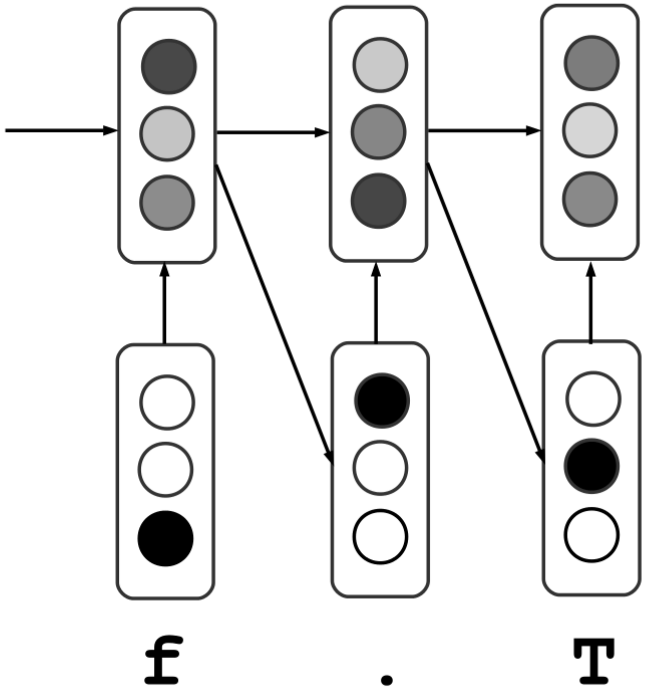
- **Supervised discriminative** model performs at state-of-the-art with little training data
- Enables easy inclusion of **external signals**

Extras

Architecture



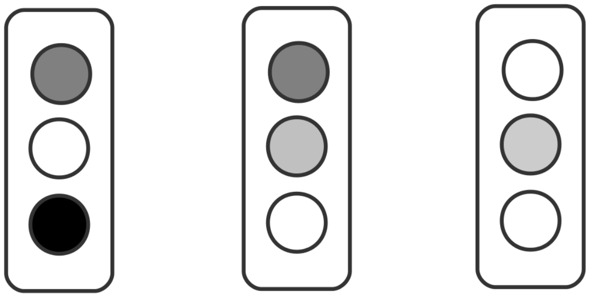
Simple Recurrent Networks



Elman, J. L. (1990). Finding structure in time. *Cognitive science*, *14*(2), 179-211.

Recurrent neural embeddings

- SRN trained to predict next character
- Representation:



- Embed string (at each position) in low-dimensional space

Visualizing embeddings

String	Nearest neighbors in 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