Dynamic EM in Neologism Evolution

Martin Emms

October 20, 2013

Semantic Neologism

semantic neologism: when an *old* word acquires a *new* usage/meaning

Semantic Neologism

semantic neologism: when an *old* word acquires a *new* usage/meaning

example *bricked*

Semantic Neologism

semantic neologism: when an *old* word acquires a *new* usage/meaning

example *bricked*

old sense: a construction process involving bricks, as in (from 2001)

> . . . In 1611 she was **bricked** into one of the rooms . . .

Semantic Neologism

semantic neologism: when an *old* word acquires a *new* usage/meaning

example *bricked*

old sense: a construction process involving bricks, as in (from 2001)

> . . . In 1611 she was **bricked** into one of the rooms . . .

recent sense: render a piece of equipment, often a phone, entirely unresponsive, as in (from 2011)

> I've tried to flash a custom ROM and now I think I've **bricked** my phone

Other examples

crawled	some kind of movement vs. traversal of www by a web-crawler
tweet	high-pitched bird noise vs. post to Twitter web-site

[1] Executed May 2013.

Other examples

crawled	some kind of movement vs. traversal of www by a web-crawler
tweet	high-pitched bird noise vs. post to Twitter web-site

can make problems for SMT when its training data pre-dates the neologism's emergence

some translations into German via Google Translate[1]:

English	German (via Google Translate)
he is a regular tweeter	*er ist ein regelmaessiger Hochtoener*
he has bricked my phone	*er hat mein Handy zugemauert*

[1] Executed May 2013.

Other examples

crawled	some kind of movement vs. traversal of www by a web-crawler
tweet	high-pitched bird noise vs. post to Twitter web-site

can make problems for SMT when its training data pre-dates the neologism's emergence

some translations into German via Google Translate[1]:

English	German (via Google Translate)
he is a regular tweeter	*er ist ein regelmaessiger Hochtoener*
he has bricked my phone	*er hat mein Handy zugemauert*

The question is:

Can semantic neologisms be detected from untagged text?

[1] Executed May 2013.

Representation and Notation

To talk about an occurrence of an ambiguous word will use:

$\mathbf{W}$:	words to left and right of a target
W_i:	i-th word in $\mathbf{W}$
Y:	year of occurrence
S:	sense of target occurrence of targets

Representation and Notation

To talk about an occurrence of an ambiguous word will use:

$\mathbf{W}$:	words to left and right of a target
W_i:	i-th word in $\mathbf{W}$
Y:	year of occurrence
S:	sense of target occurrence of targets

Eg. samples of **bricked**:

2001: ... *In 1611 she was* **bricked** *into one of the rooms* ...
2011: *I've tried to flash a custom ROM and now I think I've* **bricked** *my phone*

become instances:

$$Y = 2001, \quad S = 1, \quad \mathbf{W} = \langle L, In, 1611, she, was, into, one, of, the, rooms \rangle$$
$$Y = 2011, \quad S = 2, \quad \mathbf{W} = \langle and, now, I, think, I've, my, phone, R, R, R \rangle$$

Outline

Time dependent Sense Model

Without loss of generality, using the chain rule, we have

$$p(Y, S, \mathbf{W}) = p(Y) \times p(S|Y) \times p(\mathbf{W}|S, Y)$$

The $p(S|Y)$ term directly expresses the idea that the prevalence of a sense can vary with the year

Time dependent Sense Model

Without loss of generality, using the chain rule, we have

$$p(Y,S,\mathbf{W}) = p(Y) \times p(S|Y) \times p(\mathbf{W}|S,Y)$$

The $p(S|Y)$ term directly expresses the idea that the prevalence of a sense can vary with the year

If we now assume that $p(\mathbf{W}|S,Y) = p(\mathbf{W}|S)$ ie. $\mathbf{W}$ is conditionally independent of Y given S we get first line below

Definition (Dynamic Sense Model)

$$\begin{aligned} p(Y,S,\mathbf{W}) &= p(Y) \times p(S|Y) \times p(\mathbf{W}|S) && (1) \\ &= && (2) \end{aligned}$$

Time dependent Sense Model

Without loss of generality, using the chain rule, we have

$$p(Y, S, \mathbf{W}) = p(Y) \times p(S|Y) \times p(\mathbf{W}|S, Y)$$

The $p(S|Y)$ term directly expresses the idea that the prevalence of a sense can vary with the year

If we now assume that $p(\mathbf{W}|S, Y) = p(\mathbf{W}|S)$ ie. $\mathbf{W}$ is conditionally independent of Y given S we get first line below

Definition (Dynamic Sense Model)

$$\begin{aligned} p(Y, S, \mathbf{W}) &= p(Y) \times p(S|Y) \times p(\mathbf{W}|S) && (1) \\ &= p(Y) \times p(S|Y) \times \prod_i p(W_i|S) && (2) \end{aligned}$$

Second line above by treating $\mathbf{W}$ as 'bag of words'

Outline

If we further assume that $p(S|Y) = p(S)$ we get:

Definition (Static Sense Model)

$$p(Y, S, \mathbf{W}) = p(Y) \times p(S) \times p(\mathbf{W}|S)$$

EM training

Let θ be all parameters: $p(Y), p(S|Y), p(\mathbf{W}|S)$.

Data has no sense annotation.

EM training

Let θ be all parameters: $p(Y), p(S|Y), p(\mathbf{W}|S)$.

Data has no sense annotation. So use EM to make converging sequence of estimates

$$\theta_0 \rightarrow \ldots \rightarrow \theta_n \rightarrow \theta_{n+1} \rightarrow \ldots \rightarrow \theta_{final}$$

EM training

Let θ be all parameters: $p(Y), p(S|Y), p(\mathbf{W}|S)$.

Data has no sense annotation. So use EM to make converging sequence of estimates

$$\theta_0 \rightarrow \ldots \rightarrow \theta_n \rightarrow \theta_{n+1} \rightarrow \ldots \rightarrow \theta_{final}$$

θ_n goes to θ_{n+1} by an **E**-step, followed by a **M** step

EM training

Let θ be all parameters: $p(Y), p(S|Y), p(\mathbf{W}|S)$.

Data has no sense annotation. So use EM to make converging sequence of estimates

$$\theta_0 \to \ldots \to \theta_n \to \theta_{n+1} \to \ldots \to \theta_{final}$$

θ_n goes to θ_{n+1} by an **E**-step, followed by a **M** step

(E) *generate a virtual corpus of disambiguated instances by treating each training instance* $(Y^d, \mathbf{W}^d)$ *as standing for* all *possible completions with a sense,* $(Y^d, S, \mathbf{W}^d)$, *weighting* each *by its conditional probability* $P(S|Y^d, \mathbf{W}^d; \theta_n)$, *under current probabilities* θ_n

EM training

Let θ be all parameters: $p(Y), p(S|Y), p(\mathbf{W}|S)$.

Data has no sense annotation. So use EM to make converging sequence of estimates

$$\theta_0 \rightarrow \ldots \rightarrow \theta_n \rightarrow \theta_{n+1} \rightarrow \ldots \rightarrow \theta_{final}$$

θ_n goes to θ_{n+1} by an **E**-step, followed by a **M** step

(E) *generate a virtual corpus of disambiguated instances by treating each training instance* $(Y^d, \mathbf{W}^d)$ *as standing for* all *possible completions with a sense,* $(Y^d, S, \mathbf{W}^d)$*, weighting* each *by its conditional probability* $P(S|Y^d, \mathbf{W}^d; \theta_n)$*, under current probabilities* θ_n

(M) *apply maximum likelihood estimation to the virtual corpus to derive new estimates* θ_{n+1}.

EM update equations

For each data item d, let $\gamma^d_{\theta_n}(s)$ be **the conditional S-prob under θ_n** ie.

$$\gamma^d_{\theta_n}(s) := P(S = s | Y = y^d, \mathbf{W} = \mathbf{w}^d; \theta_n)$$

can prove the E-M cycle leads to update formulae:

EM update equations

For each data item d, let $\gamma^d_{\theta_n}(s)$ be **the conditional S-prob under θ_n** ie.

$$\gamma^d_{\theta_n}(s) := P(S = s | Y = y^d, \mathbf{W} = \mathbf{w}^d; \theta_n)$$

can prove the E-M cycle leads to update formulae:

$$P(S = s | Y = y; \theta_{n+1}) = \frac{\sum_d (\text{if } Y^d = y \text{ then } \gamma^d_{\theta_n}(s) \text{ else } 0)}{\sum_d (\text{if } Y^d = y \text{ then } 1 \text{ else } 0)}$$

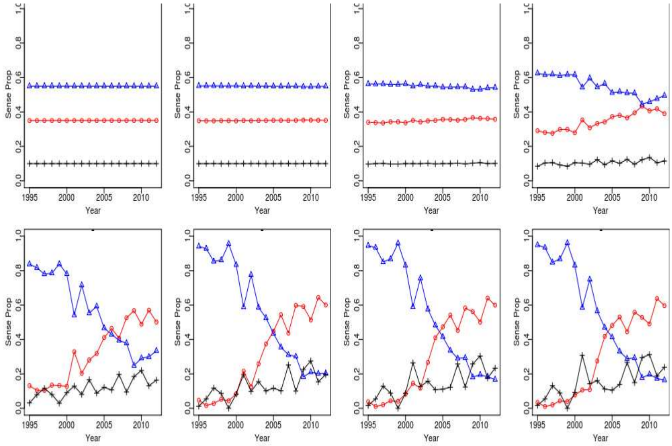
$$P(w | S = s; \theta_{n+1}) = \frac{\sum_d (\gamma^d_{\theta_n}(s) \times freq(w \in \mathbf{W}^d))}{\sum_d (\gamma^d_{\theta_n}(s) \times length(\mathbf{W}^d))}$$

Outline

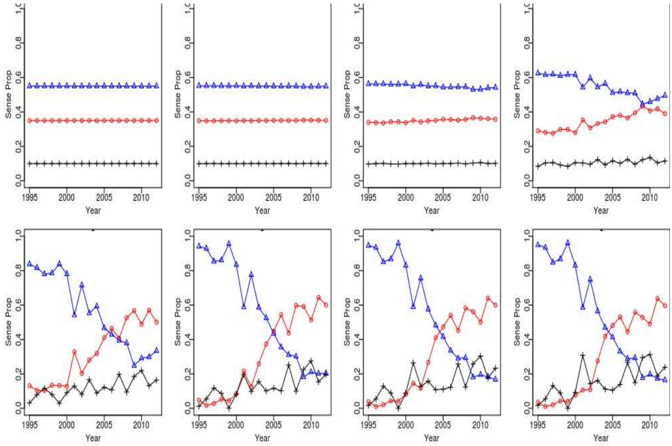
- to get time-specfic samples used the Google facility to specify a time period for searched documents
  eg. search: "bricked" 1/1/2000 – 31/12/2000
- saved 100 per year
- used window 5 words to the left of the target, and 5 words to the right
- per-sense word probs initialised to overall corpus probs + some noise
- sense distribs initialised $\frac{7}{20}, \frac{11}{20}, \frac{2}{20}$

Outline

EM converging to solution for 'crawled'

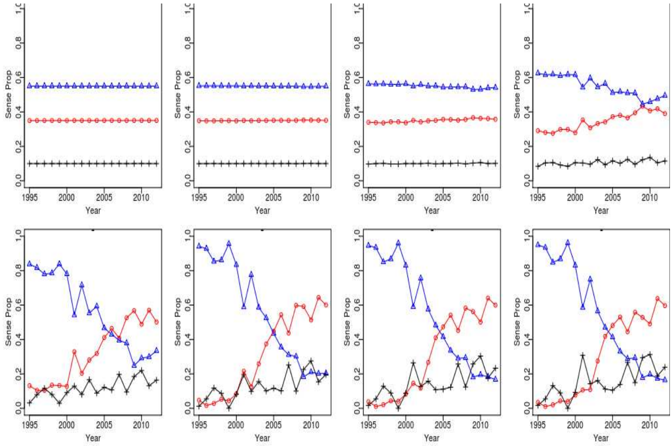


EM converging to solution for 'crawled'



red among top-20:
site : 8.64154
Google : 8.24918
pages : 6.01036
URLs : 5.8981
indexed : 4.72255
website : 4.13478
search : 3.88998
so: 'internet sense'

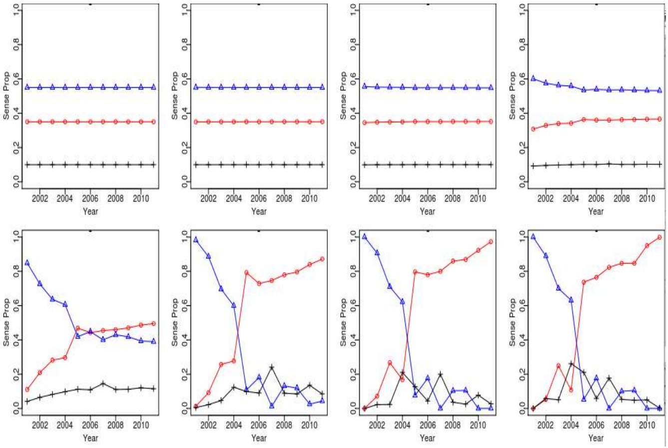
EM converging to solution for 'crawled'



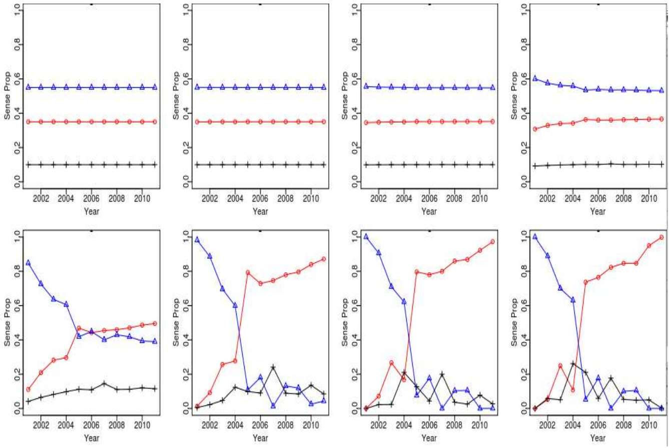
red among top-20:
site : 8.64154
Google : 8.24918
pages : 6.01036
URLs : 5.8981
indexed : 4.72255
website : 4.13478
search : 3.88998
so: 'internet sense'

blue amongst top-20:
out : 10.8425
through : 4.81634
under : 4.62561
across : 3.83244
into : 3.8
around : 3.40469
inside : 3.26545
back : 3.10985
so: 'movement sense'

EM converging to solution for 'bricked'

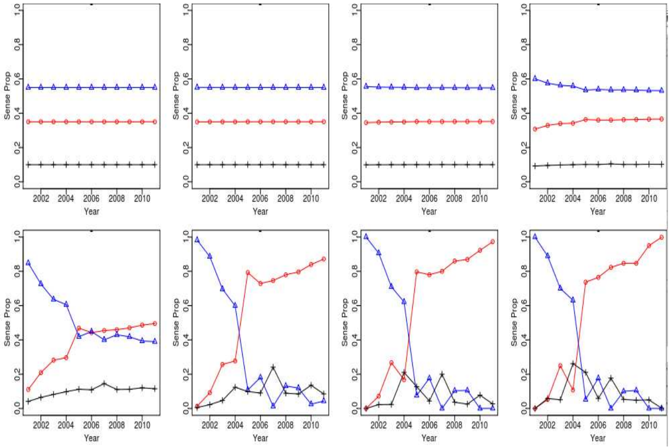


EM converging to solution for 'bricked'



red among top-20:
my
fix
iPhone
forums
Apple
firmware
Samsung
update
so: 'phone sense'

EM converging to solution for 'bricked'



red among top-20:
my
fix
iPhone
forums
Apple
firmware
Samsung
update
so: 'phone sense'

blue amongst top-20:
up
in
home
window
wall
fireplace
door
so: 'construction sense'

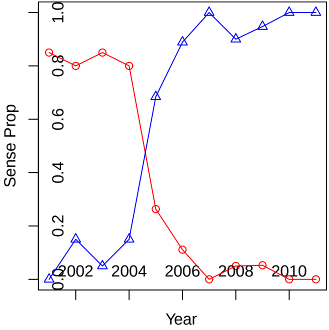
Comparing to labelled target

the algorithm learns from data with *no* sense data. For 'bricked' we hand-labelled to give a target to compare to.

Comparing to labelled target

the algorithm learns from data with *no* sense data. For 'bricked' we hand-labelled to give a target to compare to.

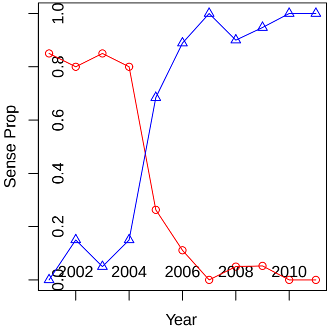
- The inferred sense distrib resembles the empirical target:



Comparing to labelled target

the algorithm learns from data with *no* sense data. For 'bricked' we hand-labelled to give a target to compare to.

- The inferred sense distrib resembles the empirical target:



- If the EM-trained models are used to label the data, then dynamic model accuracy: 82.4%.
static model accuracy: 76.1%

Conclusions and Further Directions

- some evidence that can spot a semantic neologism
- further data
- more elaborate models: prior on year-to-year change
- comparison to LDA and dynamic topic models