Conditional language models for linguistic variation and change

Bill Noble

Centre for Linguistic Theory and Studies in Probability
Institutionen för filosofi, lingvistik och vetenskapsteori
Göteborgs universitet

Computational Detection of Language Change Workshop
@ SLTC
25 November, 2020

Conditional language models

A language model estimates the probability of a sequence by predicting the next word, given the sequence so far.

$$P(w_1, ...w_n) = \prod_{i=1}^{n} P(w_i \mid w_1, ...w_{i-1}) \tag{1}$$

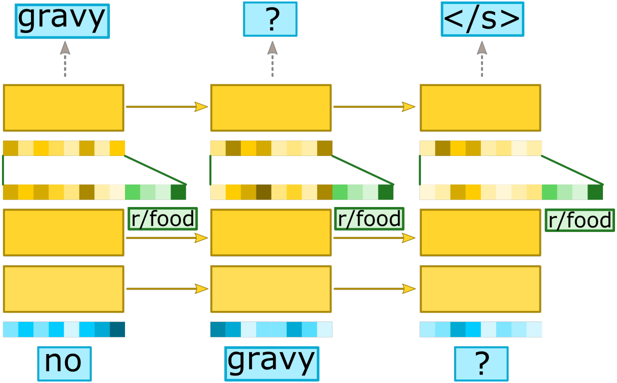
A *conditioned* language model takes additional context, c, into account.

$$P(w_1, ...w_n \mid c) = \prod_{i=1}^{n} P(w_i \mid w_1, ...w_{i-1}; c) \tag{2}$$

Conditional multi-layer neural language models

- Common neural language modelling technique: Concatenate a vector representation of c to the input
- This is commonly used in generative models to get the model to generate text that's relevant to some context, c.
  - Image captioning: concatenate image representation (e.g., Vinyals et al., 2015)
  - Machine translation: concatenate source sentence representation (e.g., Kalchbrenner and Blunsom, 2013)
- In a multi-layer model, we can also inject c between layers by concatenating it to the hidden layer

Community-conditioned language models[1]



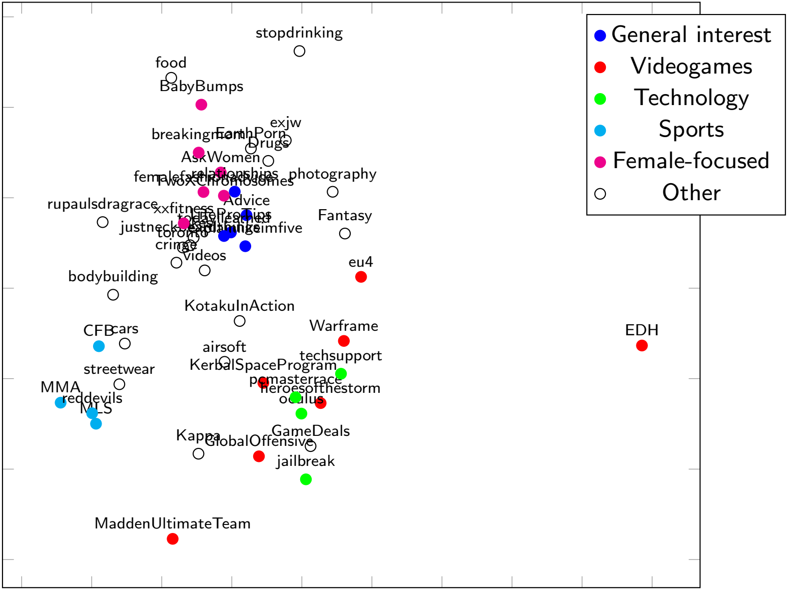
By conditioning on community, we can account for community-level linguistic variation.

[1] Joint work with Jean-Philippe Bernardy.

Conditioning on community improves LM performance

	l_c	test epoch	Perplexity	Info. gain
LSTM	-	21	51.99	-
	0	17	50.83	1.023
	1	34	49.66	1.047
	2	11	50.23	1.035
	3	16	**49.60**	**1.048**
Transformer	-	20	61.43	-
	0	7	58.71	1.046
	1	12	61.69	0.992
	2	7	78.76	0.780
	3	10	**52.28**	**1.054**

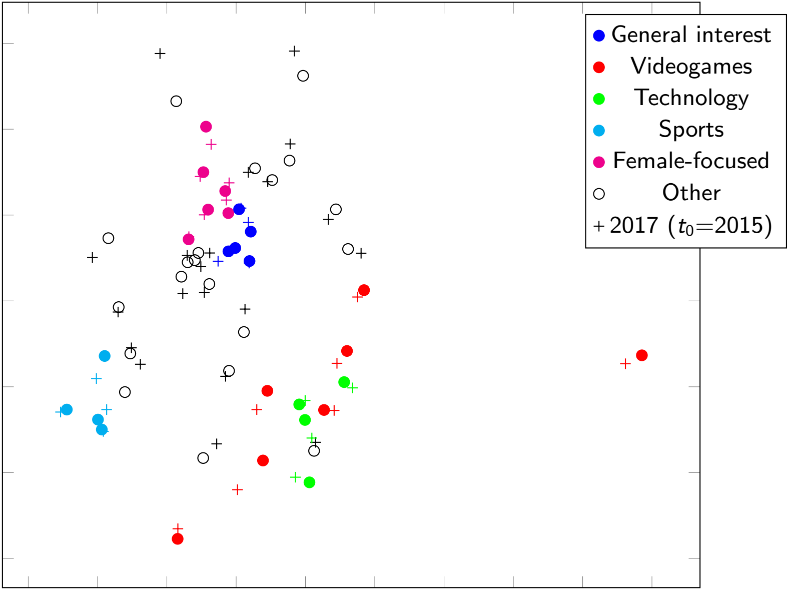
The community embedding (PCA)



Diachronic community-conditioned models: Naive approach

- Idea: Use an embedding for each community × time period
    - With 46 communities and 2 time periods (2015, 2017) we now have 92 conditional vectors.
- Concatenate the community embedding at layer 0 (i.e., directly to the word embedding)

Diachronic community embedding



Word-level change

We have:

- W : $[\texttt{vocab\_size}, \texttt{word\_emsize}]$ – word embeddings
- C : $[\texttt{n\_comms} \times \texttt{n\_time\_periods}, \texttt{cond\_emsize}]$ – conditional embedding
- A : $[\texttt{cond\_emsize} + \texttt{word\_emsize}, \texttt{word\_emsize}]$ – linear layer (before input to the LSTM)

This gives us:

- $W'_{i,j,t} = (W_i \oplus C_{(j \cdot t)}) \cdot A$ – word i “contextualized” by community j in time period t
- $\hat{W}_{i,j} = cos\ dist(W'_{i,j,1}, W'_{i,j,2})$ – community-specific lexical change

Most changed words

We consider words that changed the most in a given community, relative to the same word in other communities. In particular, we consider:

$$\frac{\hat{W}_{i,j} - \sigma_i}{\mu_i}$$

where $\mu_i = \sum_j \hat{W}_{i,j}/|C|$ and σ_i is the associated standard deviation.

Words with the highest relative change

	Advice	AskWomen	BabyBumps	CFB	Drugs
0	méxico	stock	nicks	bloatware	navy
1	rally	nicks	simulation	os	shovelware
2	stock	rally	tsunami	mbr	camo
3	name	core	rebranding	touchscreen	platinum
4	puck	xbmc	rendering	soundcard	attire

	EDH	EarthPorn	Fantasy	GameDeals	GlobalOffensive
0	suburb	scarring	forklift	mouth	crest
1	county	bravado	throttling	yak	ingenuity
2	diets	prowess	cyclone	telepathy	caviar
3	york	medic	liquid	testicle	paints
4	suburbs	rng	boop	cigar	vegemite

	Jokes	Kappa	KerbalSpaceProgram	KotakuInAction	LifeProTips
0	ovr	panhandle	prom	mush	nicks
1	5.0.1	keto	knit	shotty	finishes
2	gear	supplement	bodycon	forma	name
3	jailbreak	ingestion	jean	progress	garbage
4	blueprint	bulking	chiffon	gallium	legends

	MLS	MMA	MaddenUltimateTeam	TwoXChromosomes	Warframe
0	headspace	kit	agnosticism	shotty	lansing
1	os	magnification	doctrine	stock	crest
2	introspection	coloring	famine	xbmc	ogden
3	bloatware	bokeh	gypsies	méxico	photoshopped
4	prescription	liberation	inventions	finishes	shaven

	airsoft	bodybuilding	breakingmom	cars	cringe
0	coulter	vocabulary	ao	sylvanas	blueprint
1	deman	symbolism	rendering	blanche	base
2	intervention	libya	nicks	asd	dmr
3	intervening	croft	gear	tyrande	tek
4	pakman	fiction	leg	arthas	forma

Words with the highest relative change

	eu4	exjw	explainlikeimfive	femalefashionadvice	food
0	posture	stock	chica	relativity	untether
1	hammy	shotty	xbmc	wallbang	firmware
2	competitiveedh	gear	legends	bomb	5.0.1
3	curls	tek	willson	mal	cortana
4	biceps	sport	date	foul	ota

	heroesofthestorm	jailbreak	justneckbeardthings	oculus	pcmasterrace
0	crest	panhandle	forma	yak	hooligan
1	photoshopped	meat	progress	needles	hubris
2	rfk	sushi	stock	fisting	taboo
3	waldo	condiment	home	fatass	yak
4	seagull	pasta	wip	chirp	grade

	photography	reddevils	relationships	rupaulsdragrace	stopdrinking
0	untether	passthrough	eps	throttling	tsunami
1	ow	lightbulbs	hz	output	sr-71
2	medic	png	ftl	hz	clout
3	intervention	flac	toothpicks	polarity	glider
4	cr7	lobina	meu	500fps	divas

	streetwear	techsupport	todayilearned	toronto	videos
0	intoxication	draper	5.0.1	potatoe	ace
1	burdens	theo	ovr	classicfolders	experimentation
2	intervention	goats	playfire	voodoo	wip
3	manoeuvre	peppers	comp	vanilla	progress
4	immunity	savannah	mp	nostalgia	ovr

	xxfitness
0	nicks
1	riches
2	rally
3	swag
4	yahweh

Questions & continuations

- The community/time period embeddings seem to work, but the highest change lists don't look too good. Why?
  - H1: The model doesn't have enough parameters to adjust the word meanings, given community information.
  - H2: The community/time contextualzation operates on word *vectors*, but it should be parametrized by word *tokens*
- What (less naive) conditional architecture would better fit cognitive/interactional theories of language change?
- How does this proposal relate to prior work using language models for semantic change detection?

References I

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent Continuous Translation Models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, page 10, Seattle, Washington.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and Tell: A Neural Image Caption Generator. *arXiv:1411.4555 [cs]*.