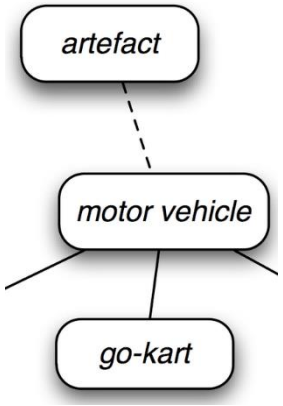
Is more better?
Is better always better?
Where's the learning?

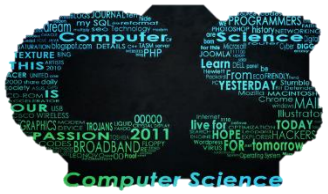
The baffling case of computational thematic fit

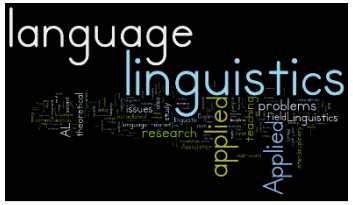
Yuval Marton

May 11, 2022

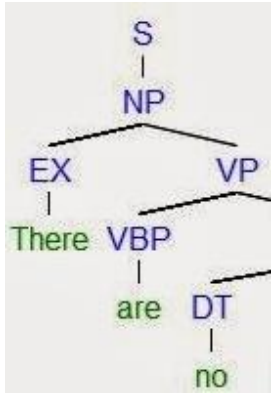


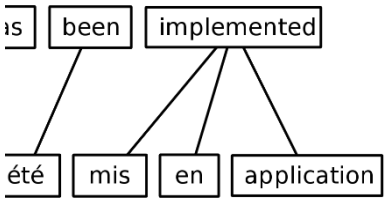
About the Speaker

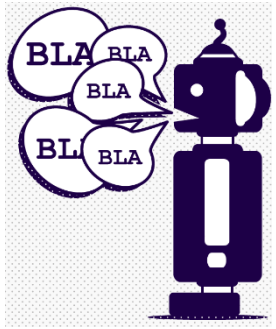


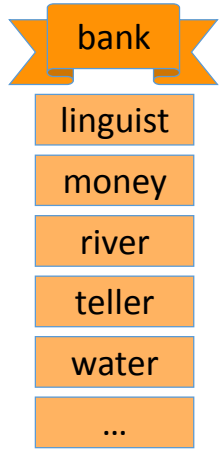


- Background: both CS and Linguistics
  Neuroscience and cognitive studies
- Affiliate Professor at UW
- Industry experience: from startups to big corporations
- Research interests and expertise areas:
  - **Semantic processing**
    - **Semantic role labeling (SRL), Thematic Fit, key-phrase/relation/fact extraction,**
    - **Lexical/word representation (learning), Semantic similarity, Paraphrasing**
  - **Syntactic parsing**
  - **Machine Translation**
  - **Chatbots / Natural Language Understanding (NLU)**
  - **Search**: ranking search results





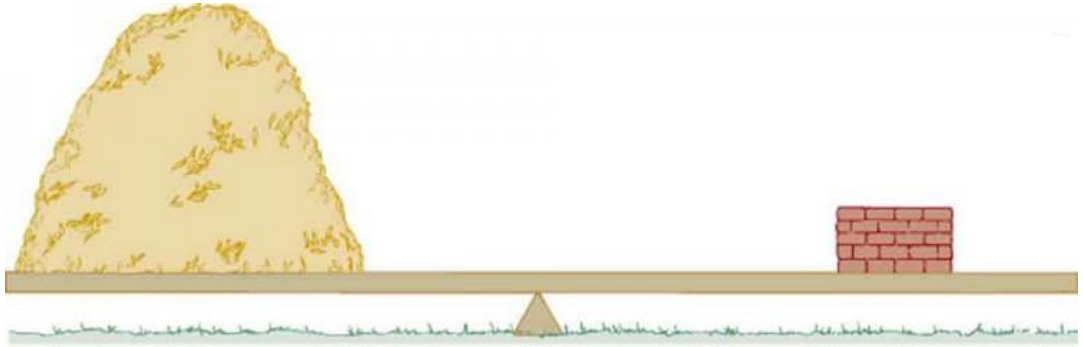




Motivation

- I wanted to improve SRL / Machine Understanding
- I tweeted about it!
- Asad Sayeed (now my co-author) pointed me to his large, balanced annotated corpus
  - Asad: “I work on thematic fit… very related to SRL…”
  - Yuval: “Psycholinguistics! Cool! Finally a justification for my NACS certificate!”
- I looked at the SRL annotation… and was appalled!
- We started ~~arguing~~ discussing whether quality matters at scale…
  - Asad: “Google!”
  - Yuval: “GIGO!”

Background



- With advent of Big-Data (and faster more powerful hardware), the received wisdom is that more (so-so) data → better models
  - More “silver” labeled data >= small expert-annotated “gold” labeled data
  - More crowdsourced lay-person labels >= small expert-annotated labels
  - Labeled data by mediocre taggers >= less labeled data by SotA taggers
  - McClosky et al. (2006), Foster et al. (2007), Petrov et al. (2010), …

Research questions (1)

- Does “more data = better” apply to
  - Semantic role prediction? (SRL “lite”: given a verb and arg, what’s its role?)
  - Role filling / word prediction? (given predicate and role)
  - **Thematic fit estimation?**
  - **Especially as a related task the model was not directly optimized for?**

- But is it always so?
  - Are there settings or tasks where more data isn’t better?
  - Are there settings or tasks where better annotations keep their advantage at scale?

- Marton and Sayeed (LREC 2022)

Terminology (1): Syntax and semantics primer

- Syntactic parses
  - Often identify argument spans, but not always (light verbs, open ngrams,…)
  - Miss generalizations such as active/passive voice
  - Brittle with domain/genre change
- Semantic parses / semantic processing
  - Offer generalizations of either semantic frames or thematic roles
  - ProbBank, VerbNet, FrameNet, AMR, …
  - SRL (syntax-based, syntax-aware, or syntax-free)
  - Typically more robust
- Combinations

Terminology (2)

Task	Input	Output	Comments
Semantic role labeling (SRL)	Sentence (& parse?)	{predicate, {arg span, role}}	Semantic frames with their trigger & args
Selectional preference (~frame)	Predicate (verb)	roles	'cut': Agent/Arg0, Patient/Arg1, Instrument/Arg3.
Role prediction ("SRL-lite")	Predicate, arg (head)	role	"child eat apple" → prob of Agent, Patient, …
Role / slot filling (word prediction)	Predicate, role	arg head (lex item)	"child eat Patient" → prob of "apple", "cake", …
Thematic fit (semantic fit)	Predicate, arg (head), role	Fit score [0..1]	"child:Agent eat dog:Patient" → low score for dog in frame+role. Few tests sets; no training data!

Cake exs

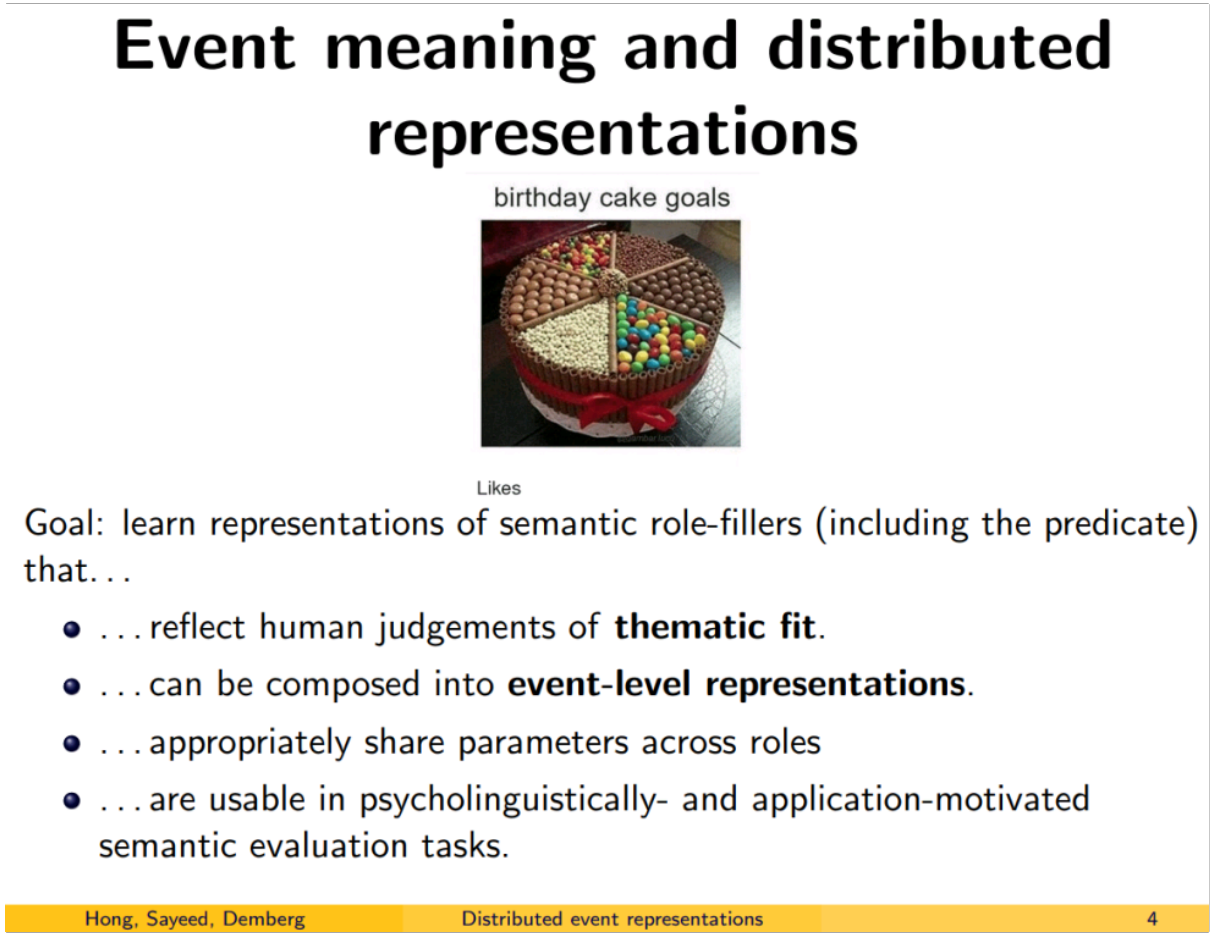
- Asad:Agent cut cake:Theme with a **knife:Instrument**
- Asad:Agent cut cake:Theme with **tongs:Instrument**
- Asad:Agent cut cake:Theme with a **flower:Instrument**
- [~~cake with flower~~]$_{NP}$

Thematic fit estimation: challenges

- Few small test sets, no large general training data!
- Related to SRL, role filling, role prediction
- Why just go for argument head?
- Do models optimized for role/word prediction learn Thematic fit too?
- Supervised previous work:
  - Padó and Lapata,2007; Herdağdelen and Baroni, 2009
- Distributional memory / neural:
  - Baroni and Lenci 2010
  - Tilk et al 2016
  - Hong et al 2018
  - Marton and Sayeed 2021/2022 (this work)

Hong et al (1)

- Inspired by Tilk et al. 2016: feed-forward single-task network
  - Observed issue: distinguish “child eats apple” from “apple eats child”
- Extended it to multi-task learning
- “child eat Patient” → probability of “apple”, “cake”, . . .
- “child eat apple” → probability of *Agent, Patient,* . . .
- Using occurrence stats, answer how common it is that this word/role is used



Hong et al (2)

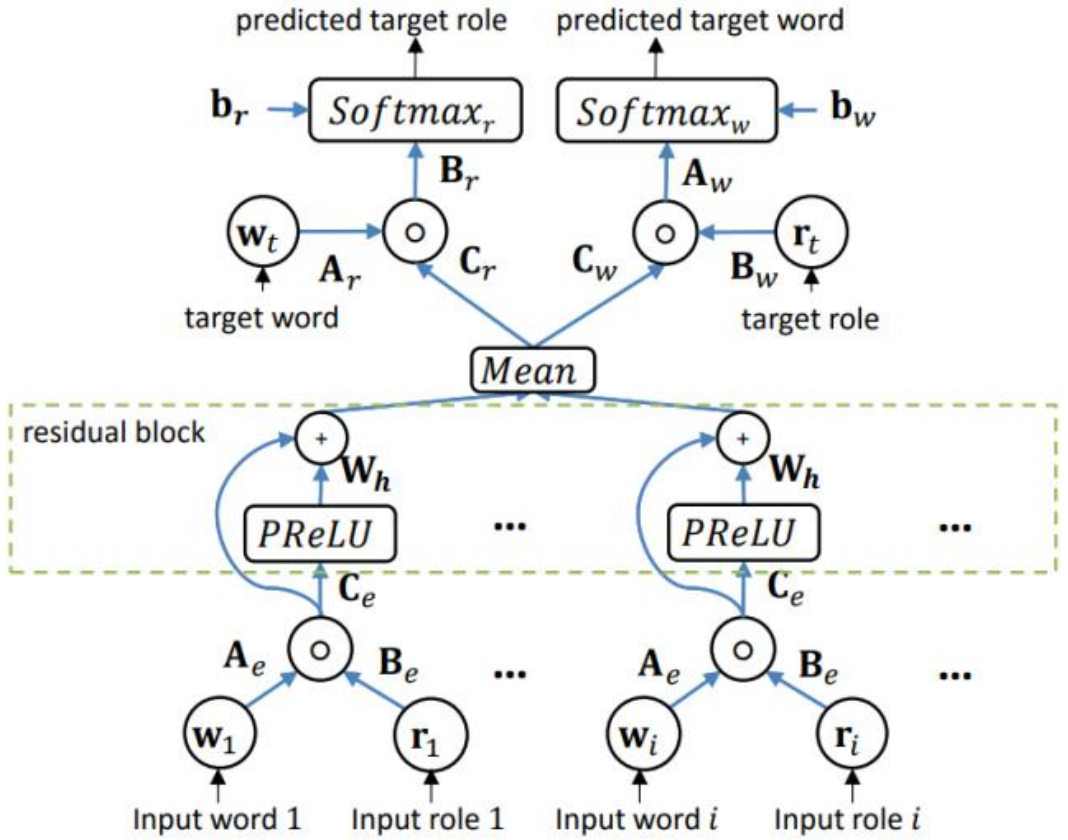
Thematic fit

Padó et al. 2007 data:

Verb	Noun	Semantic role	Score
advise	doctor	subj	6.8
advise	doctor	obj	4.0
confuse	baby	subj	3.7
confuse	baby	obj	6.0
eat	lunch	subj	1.1
eat	lunch	obj	6.9
kill	lion	subj	2.7
kill	lion	obj	4.9

New model architecture

Residual role-filler averaging model (ResRoFA-MT).



Model Thematic fit with no training data?

- Key idea: models trained for role/word prediction should have implicitly learned Thematic fit as well (at least reasonably well)
  - Optimize for both role and word prediction in multi-task learning setting
  - Big-Data (quality at/by scale)
  - GIGO? Retry with better annotations

New Lexical Resource: RW-English v2

- Created (v2 of) a large annotated corpus
  - Over 78M sentences from 2.3M docs. ~2B words (tokens), 3.5k gzip files
- v1: NLTK/WN lemmas, MaltParser, SENNA SRL (mediocre, old)
- v2: Morfette lemmas, spaCy, LSGN SRL (SotA, recent)
- **Adjustments of tokenization schemas!**
- SRL: 2 flavors
  - Syntax-aware: typically brittle out-of-domain / “in the wild”; not used here
  - Syntax-agnostic: more robust but requires syntactic head heuristics

Data split and evaluation

- Training sets: uniformly sampled subsets (0.1%, 1%, 10%, 20%)
  - Previous work: 20% “rolling window” per epoch
- Dev and test sets: 16 files (out of 3.5k) each, uniformly sampled
  - Unlike previous work (Hong et al.): last 14 files for test, last 14 before for dev
  - Role / word prediction accuracy
- Psycholinguistic (Thematic fit) test sets not optimized for:
  - Padó (2007): human-rated thematic fit score for ~400 verb-noun-role triplets, where every two triplets differ only in the role. Frequent verbs/args in WSJ
  - McRae et al. (1998): ~1500 similar triplets, less frequent (harder task)
  - Spearman’s rank correlation

Modeling

- ResNet with multi-task learning (role/word prediction)
- Role set:
  - ProbBank's PRD, Arg0, Arg1, ArgM-TMP, ArgM-LOC, ArgM-MNR
  - Arg0 ≈ agent, Arg1 ≈ patient / theme
  - Separate labels for *missing role* and *unknown role* instead of one for both
- Vocab: 50k most frequent lemma forms (same as Hong et al.)

Results on (in)directly optimized for tasks

- **v2 outperforms v1** on most subsets (few surprises), no convergence?
- **Quality improves with size** (expected), perhaps plateauing (but Padó-max)
- x2 and **up to x10 data "saving"** (except McRae)

Training sample (# trials)	Version	Directly optimized for: Role acc.	Word acc.	ρPadó final	ρPadó max	ρMcRae final	ρMcRae max
0.1% (3)	v1	.8857	.0435	.2760	.2760	.1924	.1968
	v2	.9102	.1029	.3149	.3257	.1934	.2065
1% (5)	v1	.9332	.0819	.5150	.5230	.3142	.3157
	v2	.9656	.1416	.4850	.4975	.3368	.3398
10% (3)	v1	.9419	.0941	.5166	.5368	.3996	.4126
	v2	.9715	.1541	.5229	.5623	.3935	.3981
20% (3)	v1	.9445	.0982	.5219	.5306	.4314	.4381
	v2	.9733	.1621	.5363	.5494	.4322	.4385

?

Results on (in)directly optimized for tasks

- High variance on Padó, McRae
- Clear jump in quality from 0.1% to 1%, suggesting a min: 8M sentences / 200M words for Thematic fit (& word prediction?). Perhaps a higher min for McRae (10%)

Training sample (# trials)	Version	Role acc.	Word acc.	ρPadó final	ρPadó max	ρMcRae final	ρMcRae max
0.1% (3)	v1	.8857 ±.0009	.0435 ±.0001	.2760 ±.0331	.2760 ±.0331	.1924 ±.0110	.1968 ±.0124
	v2	.9102 ±.0063	.1029 ±.0007	.3149 ±.0308	.3257 ±.0412	.1934 ±.0044	.2065 ±.0057
1% (5)	v1	.9332 ±.0006	.0819 ±.0002	.5150 ±.0299	.5230 ±.0141	.3142 ±.0079	.3157 ±.0069
	v2	.9656 ±.0001	.1416 ±.0002	.4850 ±.0135	.4975 ±.0141	.3368 ±.0130	.3398 ±.0118
10% (3)	v1	.9419 ±.0017	.0941 ±.0005	.5166 ±.0345	.5368 ±.0020	.3996 ±.0206	.4126 ±.0091
	v2	.9715 ±.0010	.1541 ±.0045	.5229 ±.0227	.5623 ±.0227	.3935 ±.0192	.3981 ±.0223
20% (3)	v1	.9445 ±.0003	.0982 ±.0011	.5219 ±.0069	.5306 ±.0073	.4314 ±.0123	.4381 ±.0032
	v2	.9733 ±.0004	.1621 ±.0048	.5363 ±.0035	.5494 ±.0111	.4322 ±.0232	.4385 ±.0257

Comparison to baseline: Hong et al. (2018)

- Our v2 max result on **Padó-all** (59.9% best single run, 56.2% averaged) **outperforms** their 53% with only 10% of the data (half theirs?)
- Our max result on **McRae-all** (45.9% best single run, 43.9% averaged) **outperforms** their 42.5%, despite less frequent pred-arg combos.
- Our v2 **role accuracy** (97.3%) **outperforms** their 94.7%, even at 1% of their training data (96.6%), presumably due to our separating *unknown role* from *missing role*.

Validating v2 quality advantage

- v2 has ~20% more semantic frames (see table)
- Qualitative (non-representative) random sample: 8 sentences (take with grain of salt!)

data set	previous (v1)	this (v2)
training 10%	16,889,581	20,151,313
dev	766,333	915,473
test	767,325	919,365

- v2 had a clear advantage over RE in correctly identifying frames, roles
  - both in #frames, #roles and #sentences with offending cases (63–75%).
  - with almost no cases in which v1 did better.
  - Both v1,v2 had few wrong roles, similar in number

Quality vs quantity

- **quality** + **quantity**:
  better argument span and role prediction in LSGN (v2) vs SENNA (v1)?
- greater **frame quantity** of predicted frames with LSGN (~20% more), compounded by
- better parsing **quality** of spaCy (v2) vs MaltParser (v1) → arg heads
- Morfette's (v2) better **quality** of lemma analysis

- Note: sentence quantity is the same in v1,v2
- Not **frame quantity** : 1%-training v2 outperformed 10%-training v1 on role and word prediction, even though the former was trained on eighth of the number of frames (and tenth of the sentences).
- Similarly with 0.1% v2 outperforming 1% v1 on word prediction.

Ethical considerations

- This work used two large corpora (the BNC and ukWaC), hence it is not practical
- to completely account for all the data in the corpus.
- The BNC is a curated corpus, but part of their transcribed conversations were recorded without prior consent of all recorded individuals. This is no longer an acceptable conduct in Great Britain and many other countries.
- Our annotated corpus clearly marks the source of each sentence, so those who wish to exclude BNC data can easily do so.
- Insofar as future work keeps that in mind, we believe there to be minimal scope for direct misuse of our results.

Role Distribution and Granularity

- Role set granularity – read in the paper!

Count	Label
2,120,947	ARG1
1,234,063	PRD
1,090,751	ARG0
688,268	ARG2
380,294	ARGM-TMP
257,056	ARGM-MOD
227,040	ARGM-ADV
220,502	ARGM-MNR
194,532	ARGM-LOC
95,724	ARGM-DIS
87,036	ARGM-NEG
68,156	ARGM-PRP
39,780	ARGM-DIR
35,938	ARGM-ADJ
31,004	ARG3
27,850	ARGM-CAU
22,092	ARG4
18,254	ARGM-EXT
13,456	ARGM-PRD
9,108	ARGM-LVB
5,540	ARGM-GOL
3,826	ARGM-COM
3,460	ARGM-PNC
1,686	ARGM-REC
12	ARG5

Table 3: SRL label counts in dev set

Research questions (2)

- Where's the learning?
  - How come random word embeddings perform as well as pre-trained ones?
  - Is (most of) the learning stored in the word emb? Role emb? "The network"?
  - The effect of (domain-general) training set size on thematic fit estimation
- With Samrat Halder and Mughil Muthupari
  - Follow-up work which started as a Columbia University Data Science Institute Capstone project 2021
- Muthupari^, Halder^, Sayeed and Marton (submitted, 2022)
  - ^ both contributed a lot

Tuning embeddings is (not really) all you need

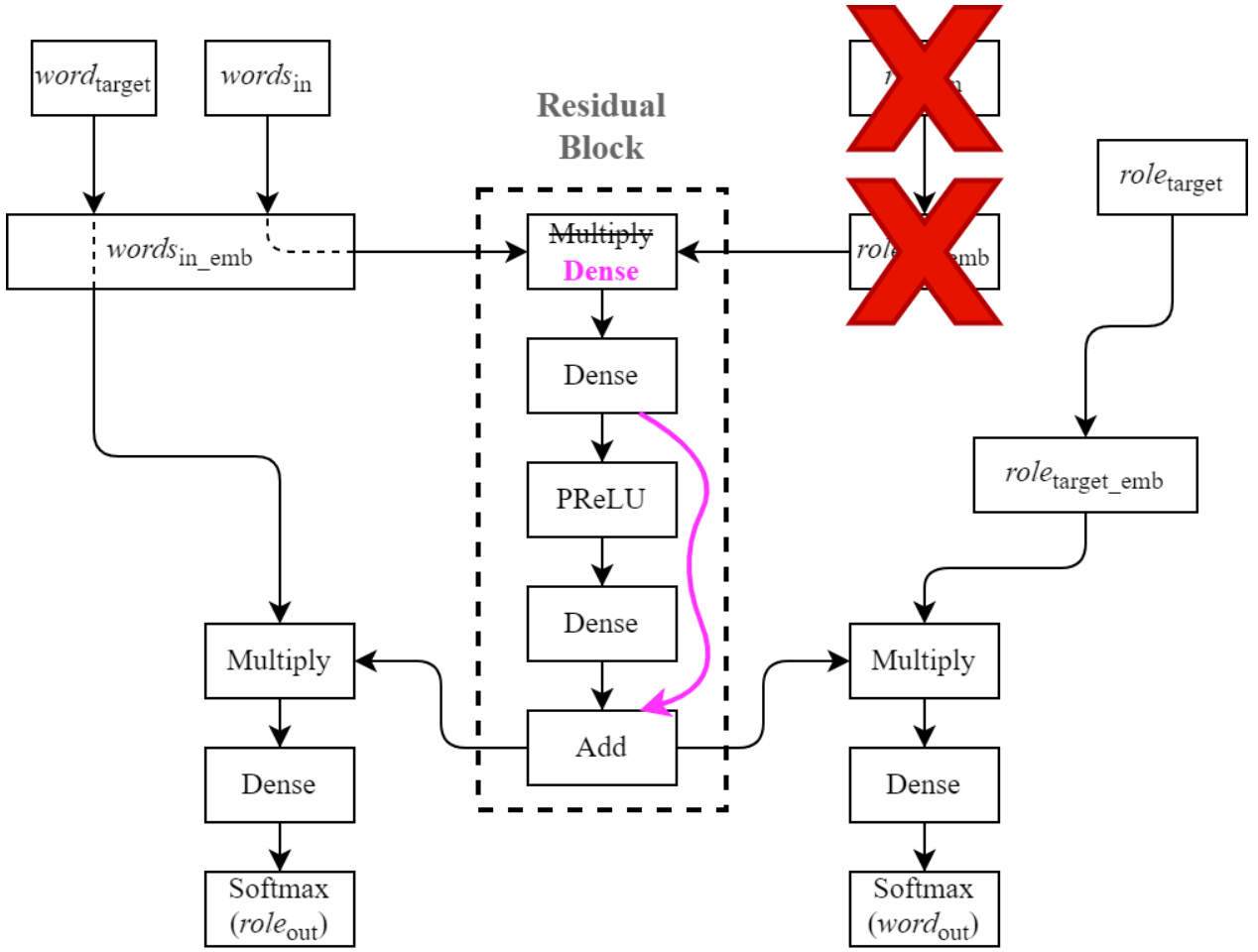
Embedding	Shared?	Tuned?	Role?	Role Accuracy	Word Accuracy	Epochs*
Random	N	Y	Y	.9655 ± .0014	.1363 ± .0020	11(6)
Random	**Y**	Y	Y	.9671 ± .0003	.1372 ± .0022	11(6)
GloVe	Y	Y	Y	.9669 ± .0003	.1374 ± .0005	15(10)
Random	Y	**N**	Y	.6609 ± .0046	.1208 ± .0012	25(20)
GloVe	Y	N	Y	.9510 ± .0011	.1291 ± .0006	25(20)

- Without tuning random is bad
- Tuning helps pre-trained too! (role/word, Padó, McRae)
- Untuned GloVe best at Ferretti location + instrument
- Bicknell near chance?

Embed.	Shrd	Tuned	Role	Padó	McRae	GDS	Ferretti-Loc	Ferretti-Instr	Bicknell
Random	N	Y	Y	.5474 ± .0345	.3231 ± .0236	.4485 ± .0314	.2611 ± .0036	.2282 ± .0623	.5260 ± .1185
Random	**Y**	Y	Y	.5280 ± .0274	.3384 ± .0174	.4388 ± .0206	.2532 ± .1421	.2266 ± .0391	.5000 ± .0673
GloVe	Y	Y	Y	.5316 ± .0320	.3280 ± .0177	.4534 ± .0209	.2851 ± .0301	.2895 ± .0258	.5438 ± .0370
Random	Y	**N**	Y	.4396 ± .0344	.2838 ± .0109	.2841 ± .0246	.1767 ± .0273	.2086 ± .0322	.4781 ± .0450
GloVe	Y	N	Y	.4941 ± .0247	.3090 ± .0254	.4349 ± .0229	.3011 ± .0301	.3439 ± .0421	.5563 ± .0490

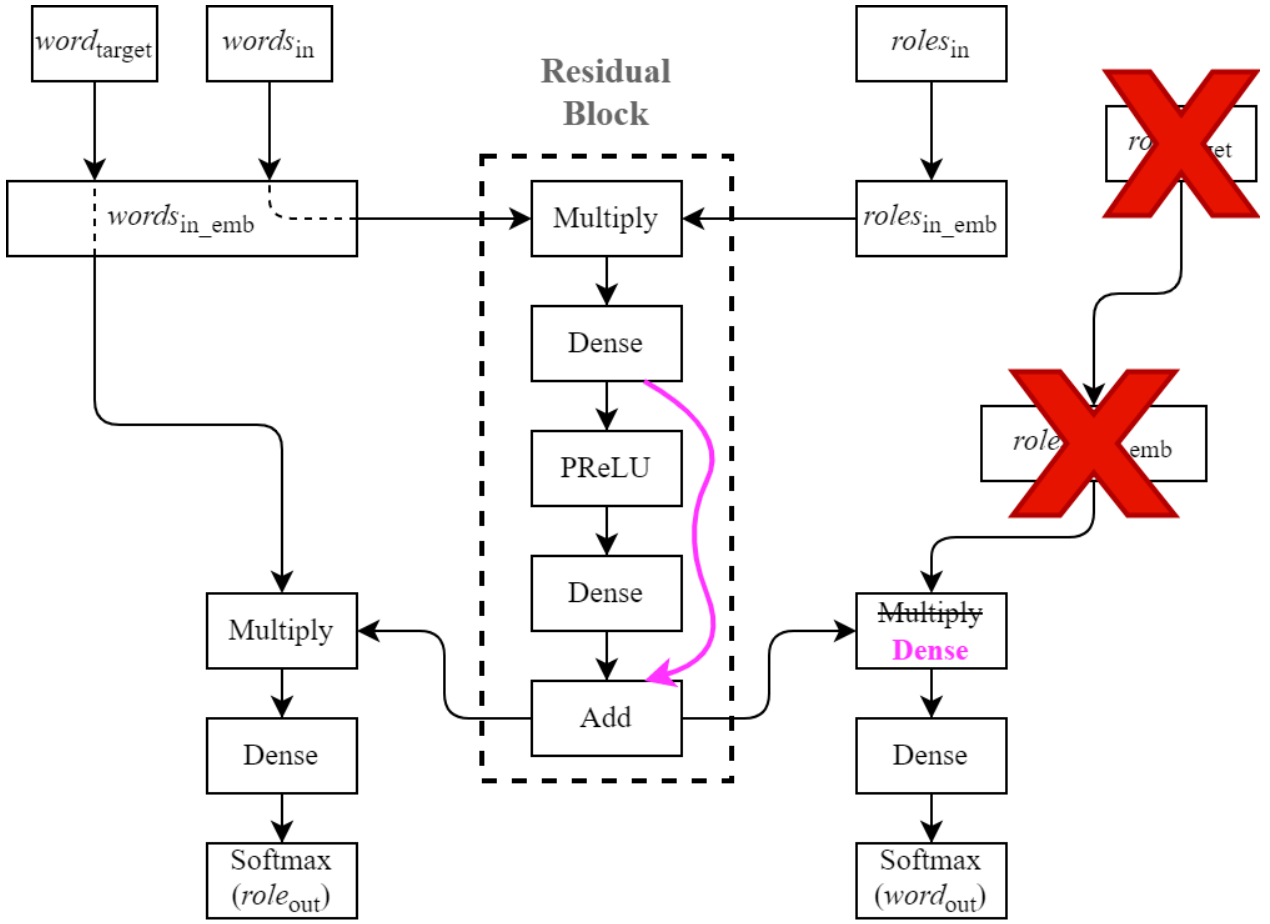
Ablation tests: No input roles (NIR)

- But note we still have role info in the target role



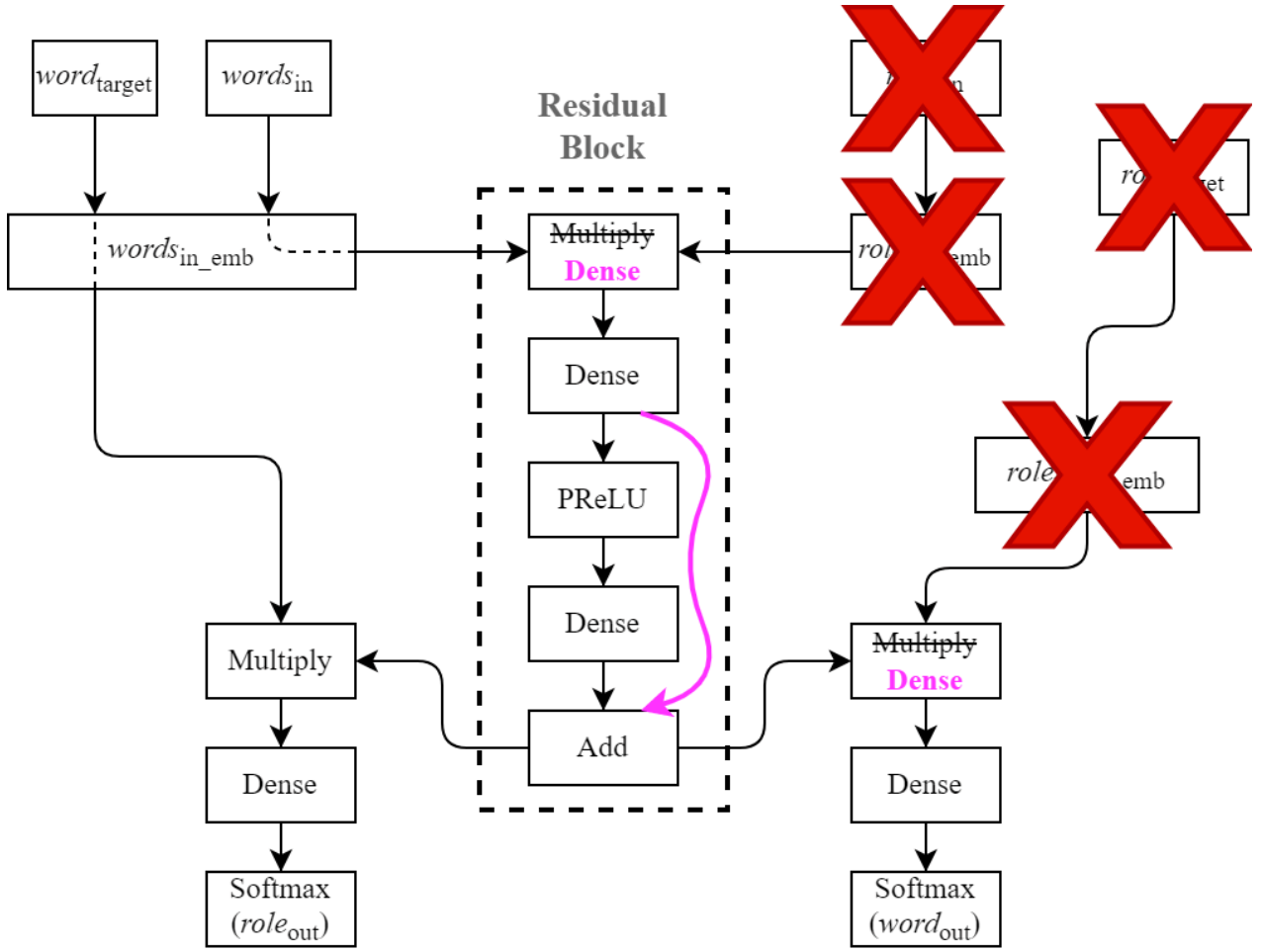
Ablation tests: No target role (NTR)

- But note we still have role info in the input roles



Ablation tests: No roles (NR)

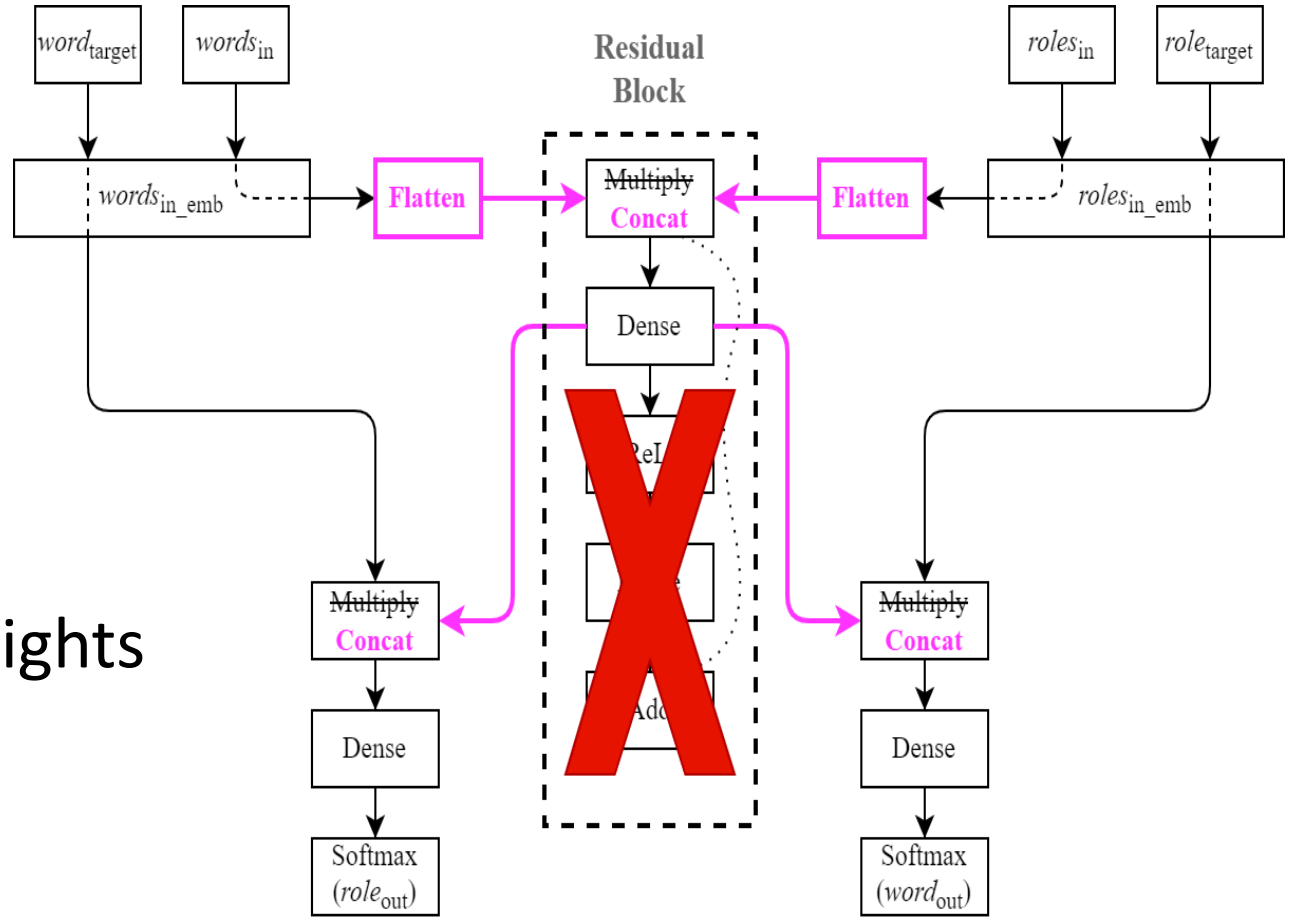
- No role info
- “NIR + NTR”



Ablation tests: No “smart network”

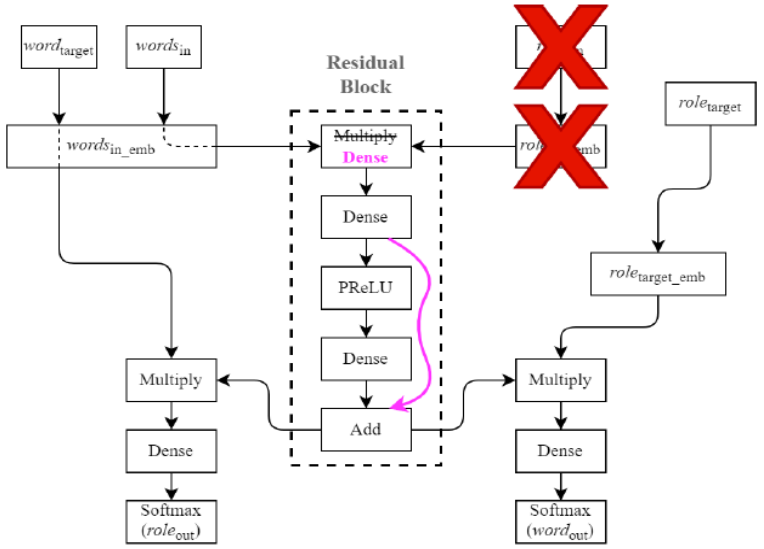
- Simple Net:
  No “smart network”
  (no word x role mult,
  no resnet)

- Random Net:
  same architecture as the
  baseline but with random weights
  instead of learned weights
  (not illustrated)

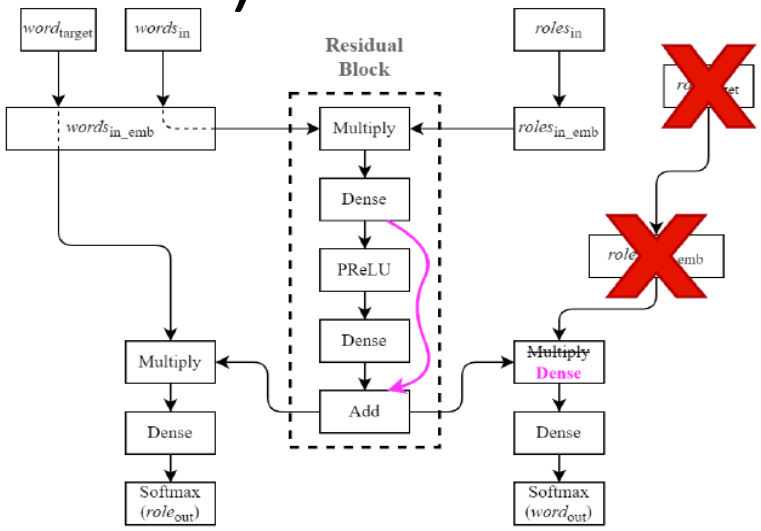


Where's the learning? (Ablation tests)

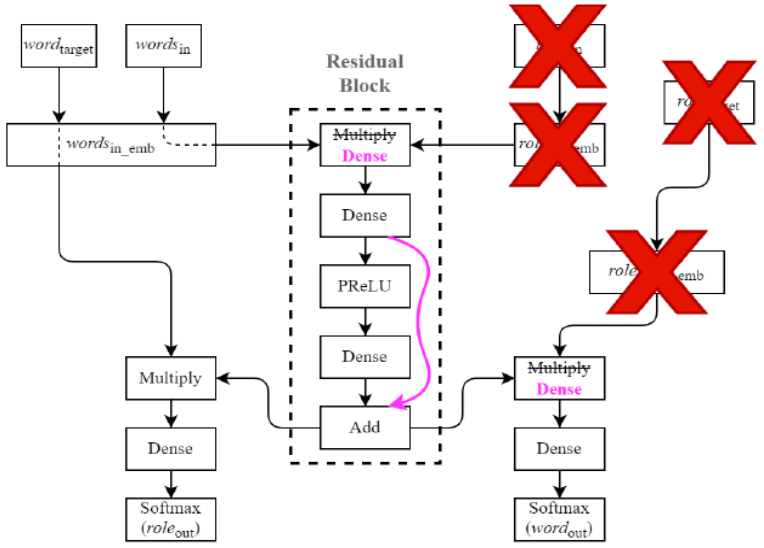
- No input roles (NIR)
- No target role (NTR)
- No roles (NR)
- No learning in network (random weights)
- No "smart network" (no word x role mult, no resnet)
- Random Net: random weights (not illustrated)



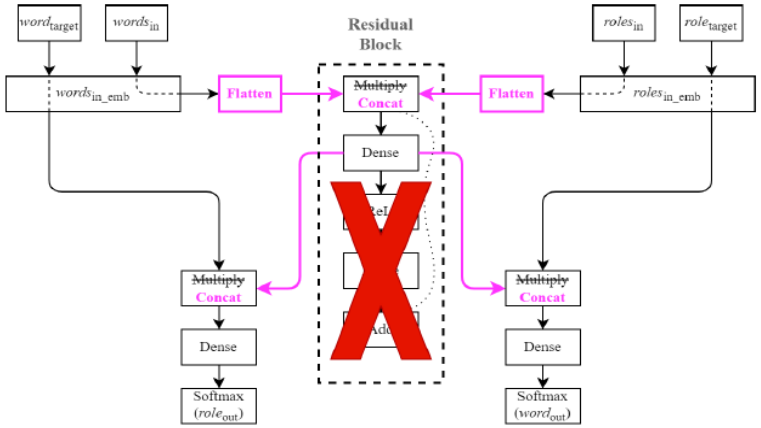
(c) **NIR** network



(d) **NTR** network



(e) **NR** network



(f) Simple network

Where's the learning? (Ablation tests: roles)

Embedding	Shared?	Tuned?	Role?	Role Accuracy	Word Accuracy	Epochs*
Random	N	Y	Y	.9655 ± .0014	.1363 ± .0020	11(6)
Random	**Y**	Y	Y	.9671 ± .0003	.1372 ± .0022	11(6)
GloVe	Y	Y	Y	.9669 ± .0003	.1374 ± .0005	15(10)
Random	Y	**N**	Y	.6609 ± .0046	.1208 ± .0012	25(20)
GloVe	Y	N	Y	.9510 ± .0011	.1291 ± .0006	25(20)
GloVe	Y	Y	**NIR**†	.9036 ± .0013	.1348 ± .0019	11(6)
GloVe	Y	Y	**NTR**†	.9677 ± .0006	.1230 ± .0017	12(7)
GloVe	Y	Y	**NR**†	.9007 ± .0021	.1078 ± .0010	8(3)

- Input roles: not that useful… (& hurts Bicknell?)
- Target role: crucial (except Ferretti, Bicknell?)
- Any role info: crucial (similar but also for word acc)

Embed.	Shrd	Tuned	Role	Padó	McRae	GDS	Ferretti-Loc	Ferretti-Instr	Bicknell
Random	N	Y	Y	.5474 ± .0345	.3231 ± .0236	.4485 ± .0314	.2611 ± .0036	.2282 ± .0623	.5260 ± .1185
Random	**Y**	Y	Y	.5280 ± .0274	.3384 ± .0174	.4388 ± .0206	.2532 ± .1421	.2266 ± .0391	.5000 ± .0673
GloVe	Y	Y	Y	.5316 ± .0320	.3280 ± .0177	.4534 ± .0209	.2851 ± .0301	.2895 ± .0258	.5438 ± .0370
Random	Y	**N**	Y	.4396 ± .0344	.2838 ± .0109	.2841 ± .0246	.1767 ± .0273	.2086 ± .0322	.4781 ± .0450
GloVe	Y	N	Y	.4941 ± .0247	.3090 ± .0254	.4349 ± .0229	.3011 ± .0301	.3439 ± .0421	.5563 ± .0490
GloVe	Y	Y	**NIR**	.5079 ± .0587	.3205 ± .0580	.4217 ± .0472	.3054 ± .0791	.2543 ± .0796	.6042 ± .0896
GloVe	Y	Y	**NTR**	.2400 ± .0294	.0937 ± .0258	.3845 ± .0083	.3071 ± .0017	.2621 ± .0531	.5469 ± .0388
GloVe	Y	Y	**NR**	.2496 ± .1088	.1139 ± .0150	.3385 ± .0363	.2955 ± .1243	.2668 ± .0375	.5885 ± .0448

Where's the learning? (Ablation tests: net)

Embedding	Shared?	Tuned?	Role?	Role Accuracy	Word Accuracy	Epochs*
Random	N	Y	Y	$.9655 \pm .0014$	$.1363 \pm .0020$	11(6)
Random	**Y**	Y	Y	$.9671 \pm .0003$	$.1372 \pm .0022$	11(6)
GloVe	Y	Y	Y	$.9669 \pm .0003$	$.1374 \pm .0005$	15(10)
Random	Y	**N**	Y	$.6609 \pm .0046$	$.1208 \pm .0012$	25(20)
GloVe	Y	N	Y	$.9510 \pm .0011$	$.1291 \pm .0006$	25(20)
GloVe	Y	Y	**NIR**†	$.9036 \pm .0013$	$.1348 \pm .0019$	11(6)
GloVe	Y	Y	**NTR**†	$.9677 \pm .0006$	$.1230 \pm .0017$	12(7)
GloVe	Y	Y	**NR**†	$.9007 \pm .0021$	$.1078 \pm .0010$	8(3)
RAND Network‡	Y	Y	Y	$.1530 \pm .0716$	$.0000 \pm .0000$	-
Simpler Network+	Y	N	Y	$.9987 \pm .0005$	$.1208 \pm .0020$	6(1)

- Random net: terrible, but may not be compatible with our tuned emb's
- Simple Net: reasonably bad, but best at role acc!

Embed.	Shrd	Tuned	Role	Padó	McRae	GDS	Ferretti-Loc	Ferretti-Instr	Bicknell
Random	N	Y	Y	$.5474 \pm .0345$	$.3231 \pm .0236$	$.4485 \pm .0314$	$.2611 \pm .0036$	$.2282 \pm .0623$	$.5260 \pm .1185$
Random	**Y**	Y	Y	$.5280 \pm .0274$	$.3384 \pm .0174$	$.4388 \pm .0206$	$.2532 \pm .1421$	$.2266 \pm .0391$	$.5000 \pm .0673$
GloVe	Y	Y	Y	$.5316 \pm .0320$	$.3280 \pm .0177$	$.4534 \pm .0209$	$.2851 \pm .0301$	$.2895 \pm .0258$	$.5438 \pm .0370$
Random	Y	**N**	Y	$.4396 \pm .0344$	$.2838 \pm .0109$	$.2841 \pm .0246$	$.1767 \pm .0273$	$.2086 \pm .0322$	$.4781 \pm .0450$
GloVe	Y	N	Y	$.4941 \pm .0247$	$.3090 \pm .0254$	$.4349 \pm .0229$	$.3011 \pm .0301$	$.3439 \pm .0421$	$.5563 \pm .0490$
GloVe	Y	Y	**NIR**	$.5079 \pm .0587$	$.3205 \pm .0580$	$.4217 \pm .0472$	$.3054 \pm .0791$	$.2543 \pm .0796$	$.6042 \pm .0896$
GloVe	Y	Y	**NTR**	$.2400 \pm .0294$	$.0937 \pm .0258$	$.3845 \pm .0083$	$.3071 \pm .0017$	$.2621 \pm .0531$	$.5469 \pm .0388$
GloVe	Y	Y	**NR**	$.2496 \pm .1088$	$.1139 \pm .0150$	$.3385 \pm .0363$	$.2955 \pm .1243$	$.2668 \pm .0375$	$.5885 \pm .0448$
RAND	Y	Y	Y	$-.0001 \pm .1090$	$.0109 \pm .1604$	$.0365 \pm .0784$	$.0165 \pm .1048$	$-.0346 \pm .0785$	$.4531 \pm .1027$
Simpler	Y	N	Y	$.3271 \pm .0555$	$.2175 \pm .0294$	$.3356 \pm .0345$	$.1055 \pm .0259$	$.0459 \pm .1239$	$.5365 \pm .0593$

Where's the learning? (summary)

- In (tuned / not tuned) (random / pre-trained) word embeddings?
- In role info? Elsewhere in the network?
- **The answer is task-dependent!**
  - Padó and McRae are most sensitive to ablated roles;
  - GDS, and perhaps also Bicknell, to non-tuned random word embeddings;
  - Ferretti to ablated (simplified) networks;
  - and all are sensitive to RAND Networks (duh!)

Training set size effects

Sys	Role Accuracy	Word Accuracy	Epochs
B1†	.9470	-	-
B2‡	.9715 ± .0010	.1541 ± .0045	-
20%M+	.9707 ± .0002	.1450 ± .0004	-
0.1%	.9446 ± .0015	.0994 ± .0024	12(7)
1%	.9669 ± .0003	.1374 ± .0005	15(10)
10%	.9701 ± .0002	.1443 ± .0006	13(10)
20%	.9703 ± .0004	.1445 ± .0009	9(6)
40%	.9704 ± .0007	.1442 ± .0011	9(6)
100%[2]	.9708 ± .0006	.1444 ± .0019	7(4)

- U-shape in performance as training size increases?
- (but high variance)

System	Padó	McRae	GDS	Ferretti-Loc	Ferretti-Instr	Bicknell
B1	.5300	.4250	.6080	.4630	.4770	.7450
B2	.5363 ± .0035	.4322 ± .0232	-	-	-	-
20%M	.5855 ± .0101	.4338 ± .0181	.5495 ± .0220	.3539 ± .0239	.4255 ± .0210	.6094 ± .0000
0.1%	.2992 ± .0441	.1856 ± .0157	.1699 ± .0180	.0891 ± .0306	.0367 ± .0203	.4906 ± .0402
1%	.5316 ± .0320	.3280 ± .0177	.4534 ± .0209	.2851 ± .0301	.2895 ± .0258	.5438 ± .0370
10%	.5572 ± .0247	.3993 ± .0137	.5409 ± .0150	.3410 ± .0358	.3765 ± .0320	.5906 ± .0320
20%	.5241 ± .0558	.3708 ± .1182†	.5245 ± .0148	.3191 ± .0312	.3853 ± .0454	.5813 ± .0210
40%	.3662 ± .1355	.3831 ± .0276	.5467 ± .0183	.3331 ± .0215	.3660 ± .0284	.5750 ± .0460
100%‡	.3375 ± .7293	.3733 ± .5203	.5338 ± .1328	.2736 ± .7846	.3416 ± .3297	.6094 ± .1985

Summary (1)

- Releasing large lex resource with modern linguistic annotation: **RW-Eng v2**
  - **http://yuvalmarton.com/RW-Eng**
- More (so-so) data → better models? Not always! Models trained on better lemmas, syntactic parses, and SRL tags (our **v2**) did better than v1 models
  - at all training set sizes, and even at scale
  - on both (directly supervised) role and word prediction.
  - No clear gain on (indirectly supervised) psycholinguistic tasks (except Padó large sets)
- "training dataset savings" potential:
  - training on smaller sets with better annotations yielded better results than training on datasets with less advanced annotations that were several times larger in size.
- Analyzed contributions of annotation quality and quantity & their interplay.
  - We further teased apart sentence quantity from frame quantity.

Summary (2)

- Tuning embeddings always helps! No. (often helps, not always)
- Different tasks rely mostly on different parts of the model. Why?
- U-shape in performance as training size increases? Why?
- high variance in thematic fit tasks!

Future Work

- High variance in thematic fit estimation – why? Can we stabilize it?
- Plateau and perhaps deterioration on thematic fit as training set increases. Why?
- Surprising interactions of role granularity (role set size) on these tasks. Why?
- How do word/role prediction and thematic fit tasks relate to each other?
- Add new annotation layers
- Add more evaluation tasks
- Experiment with different model architectures and loss functions

Thanks!

Thanks

- Samrat Halder, Mughil Muthupari
- Columbia DSI Capstone students
- A. Sayeed's grant: The Swedish Research Council (VR) grant (2014-39) for the Centre for Linguistic Theory and Studies in Probability (CLASP)