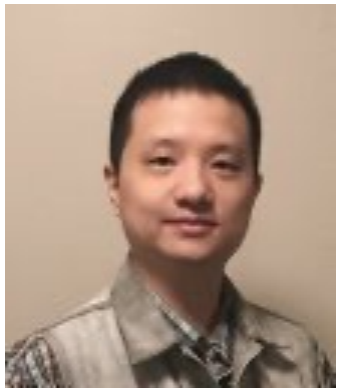
Training on Lexical Resources

Kenneth Church, Xingyu Cai, Yuchen Bian

Baidu, Sunnyvale, USA


URLs: https://github.com/kwchurch

- Syn_ant: https://github.com/kwchurch/syn_ant
  - Closely follows this paper
- ACL2022_deepnets_tutorial: https://github.com/kwchurch/ACL2022_deepnets_tutorial
  - Part A: Glass is Half-Full
    - Deep Nets can do much
  - Part B: Glass is Half-Empty
    - There is always more to do
- gft: https://github.com/kwchurch/gft
  - General Fine-Tuning: A Little Language for Deep Nets
  - Code (plus hundreds of examples)

gft (general fine-tuning):
A Little Language for Deep Nets
(Unix Philosophy: *Less is More*)

Standard 3-Step Recipe

Step	gft Support	Description	Time	Hardware
1		Pre-Training	Days/Weeks	Large GPU Cluster
2	*gft_fit*	Fine-Tuning	Hours/Days	1+ GPUs
3	*gft_predict*	Inference	Seconds/Minutes	0+ GPUs

- Terminology borrowed from *sklearn*:
  - *fit*: $f_{pre} + data \rightarrow f_{post}$
  - *predict*: $f(x) \rightarrow \hat{y}$
- *fit* and *predict* are (almost) all you need
  - *gft* programs are short (1-line)
  - No (not much) programming required
    - No python in this tutorial
    - Examples on hubs are (unnecessarily) long/complicated

Examples of 1-line GFT Programs

Step 2: *gft_fit*

```
gft_fit --eqn 'classify: label ~ text' \
    --model H:bert-base-cased \
    --data H:emotion \
    --output_dir $outdir
```

Data

f_{pre}: Pre-trained Model

f_{post}: Post-trained Model

Step 3: *gft_predict*

```
# text-classification: sentiment analysis
echo 'I love you.' | gft_predict --task text-classification
# I love you.   POSITIVE    0.9998705387115479
```

x $\hat{y}$ Score

Agenda

- Syn/Ant Binary Classification
- From Words to Texts
  - MWEs: Multiword Expressions
  - OOVs: Out of Vocabulary words
  - Multi-Lingual
  - Negation
- Leakage with Standard Benchmarks
- VAD Regression
  - VAD = Valance, Arousal, Dominance

Training on Lexical Resources

- We do not normally think of lexical resources as training data,
  - though others have trained on dictionaries
  - (Brown et al., 1993; Chairatanakul et al., 2021).
- Suppose we have a thesaurus such as (Fallows, 1898).
- Consider the thesaurus to be a set of triples: $\langle w_1, w_2, rel \rangle$
  - where w_1 and w_2 are two words,
  - and rel is 0 if w_1 and w_2 are synonyms and 1 if they are antonyms
- We can then fine-tune a pretrained deep net such as
    - BERT (Devlin et al., 2019) or
    - ERNIE (Sun et al., 2020)
  - with: $rel \sim w_1 + w_2$

Training on Fallows Thesaurus

Training (fit)

classify: gold ~ word1 + word2

0 → Synonym
1 → Antonym

word1	word2	gold
ancient	oldfashioned	0
blame	disapprove	0
clearly	confusedly	1
debt	liability	0
demure	modest	0
profitable	fruitless	1
revelry	orgies	0
rotation	order	0
vanity	selfdistrust	1

$$y \sim text_1 + text_2$$

Inference (predict)

$text_1$	$text_2$	y_1	y_2
good	bad	-3.95	4.54
bad	evil	4.44	-5.00
good	benevolent	4.43	-5.05
bad	benevolent	-3.44	4.16
good	terrorist	-3.43	4.10
bad	terrorist	4.48	-5.10

Table 1: Inference: synonymy iff $y_1 > y_2$

Data from Fallows (1898)

From Words to Text
(MWEs: Multiword Expressions)

Inference on Words

$text_1$	$text_2$	y_1	y_2
good	bad	-3.95	4.54
bad	evil	4.44	-5.00
good	benevolent	4.43	-5.05
bad	benevolent	-3.44	4.16
good	terrorist	-3.43	4.10
bad	terrorist	4.48	-5.10

Table 1: Inference: synonymy iff $y_1 > y_2$

Inference on Texts (MWEs)

$text_1$	$text_2$	y_1	y_2
freedom fighter	good	2.33	-2.56
freedom fighter	bad	-1.50	2.19
white supremacist	good	-2.05	2.91
white supremacist	bad	1.67	-1.61

Table 2: Mutiword Expressions (MWEs)

Methods

Methods

- Baselines
  - MoE: Mixture of Experts
    - Nguyen et al (2017)
    - with default settings
  - MoE with DLCE embeddings
    - Nguyen et al (2017)
    - with better settings
- Proposed Method
  - gft_fit with f_{pre} = bert (uncased)
  - gft_predict

Datasets

Fallows (1898)

Dataset	train	val	test
adj	5562	398	1986
noun	2836	206	1020
verb	2534	182	908
fallows	58,494	7190	7366
fallows-s	5886	753	777

Table 3: Sizes (edges) of synonym-antonym datasets

Baseline Results
(MoE with default settings)

Sizes of 5 Datasets

Dataset	train	val	test
adj	5562	398	1986
noun	2836	206	1020
verb	2534	182	908
fallows	58,494	7190	7366
fallows-s	5886	753	777

Table 3: Sizes (edges) of synonym-antonym d

Train on Dataset[i]; Test on Dataset[j]

Test	Train				
	adj	noun	verb	fallows	fallows-s
adj	**0.886**	0.598	0.652	0.820	0.727
noun	0.662	**0.863**	0.685	0.706	0.638
verb	0.580	0.673	**0.899**	0.820	0.731
fallows	0.621	0.566	0.556	**0.663**	0.595
fallows-s	0.629	0.574	0.537	**0.660**	0.586

Table 4: Performance (accuracy) of Mixture of Experts (MoE) with default settings. (Chance is 0.5.)

Nguyen et al (2017) report results for 3 of 25 cases

Exception to Diagonal

More Results

MoE with better settings

Test	Train				
	adj	noun	verb	fallows	fallows-s
adj	**0.921**	0.859	0.852	0.897	0.868
noun	0.841	**0.917**	0.857	0.828	0.785
verb	0.813	0.829	**0.903**	0.851	0.794
fallow	0.633	0.604	0.620	**0.666**	0.634
fallow-s	**0.659**	0.602	0.591	**0.659**	0.627

Table 5: Accuracy of MoE with dLCE embeddings.

Proposed: Fine-Tuning

Test	Train			
	adj	noun	verb	fallows
adj	**0.908**	0.657	0.713	0.881
noun	0.773	**0.877**	0.792	0.797
verb	0.767	0.722	**0.906**	0.867
fallows	0.722	0.610	0.698	**0.947**

Table 6: Accuracy with fine-tuning (bert-base-uncased).

Proposed is better on Fallows

Baseline is slightly better on datasets it was designed for

Delta (Proposed – MoE)

Test	Train			
	adj	noun	verb	fallows
adj	-0.013	**-0.202**	-0.139	-0.016
noun	-0.068	-0.040	-0.065	-0.031
verb	-0.046	-0.107	0.003	0.016
fallows	0.089	0.006	0.078	**0.281**

Table 7: Comparison of proposed method and MoE. Difference between two previous tables. MoE is better when difference is negative, and otherwise, proposed method is better. Large differences are shown in **red**.

Proposed: Fine-Tuning

Test	Train			
	adj	noun	verb	fallows
adj	**0.908**	0.657	0.713	0.881
noun	0.773	**0.877**	0.792	0.797
verb	0.767	0.722	**0.906**	0.867
fallows	0.722	0.610	0.698	**0.947**

Table 6: Accuracy with fine-tuning (bert-base-uncased).

Agenda

✓Syn/Ant Binary Classification

➢**From Words to Texts**

- MWEs: Multiword Expressions
- OOVs: Out of Vocabulary words
- Multi-Lingual
- Negation

- Leakage with Standard Benchmarks
- VAD Regression
  - VAD = Valance, Arousal, Dominance

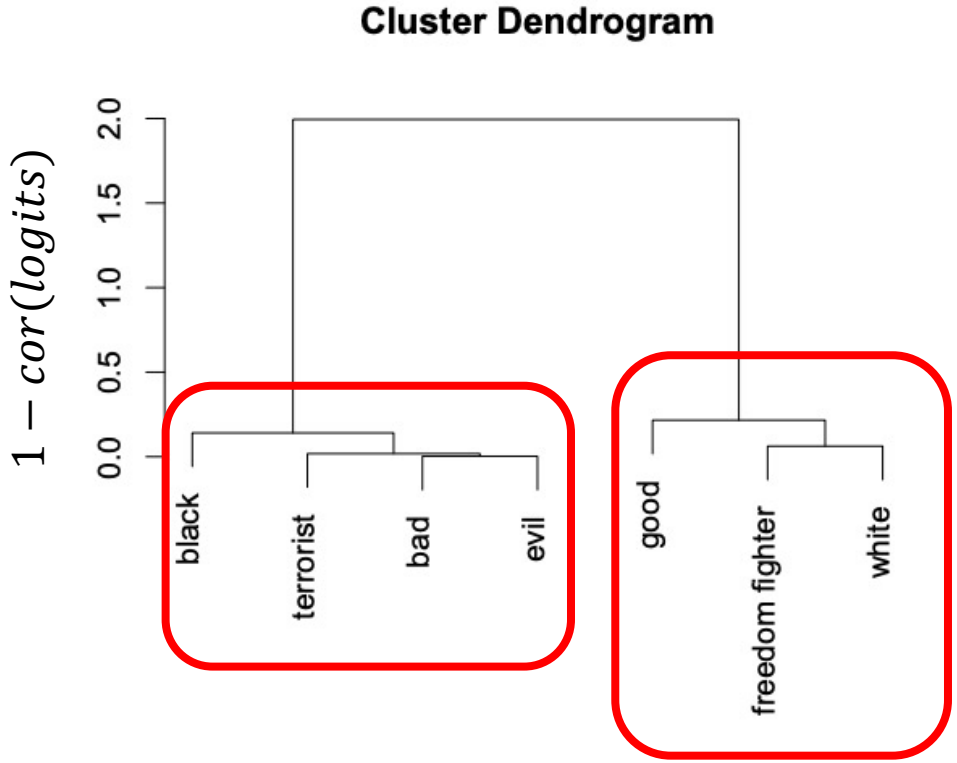
From Words to Text (Undesirable Biases)

Logits (top); Cor of Logits (bottom)

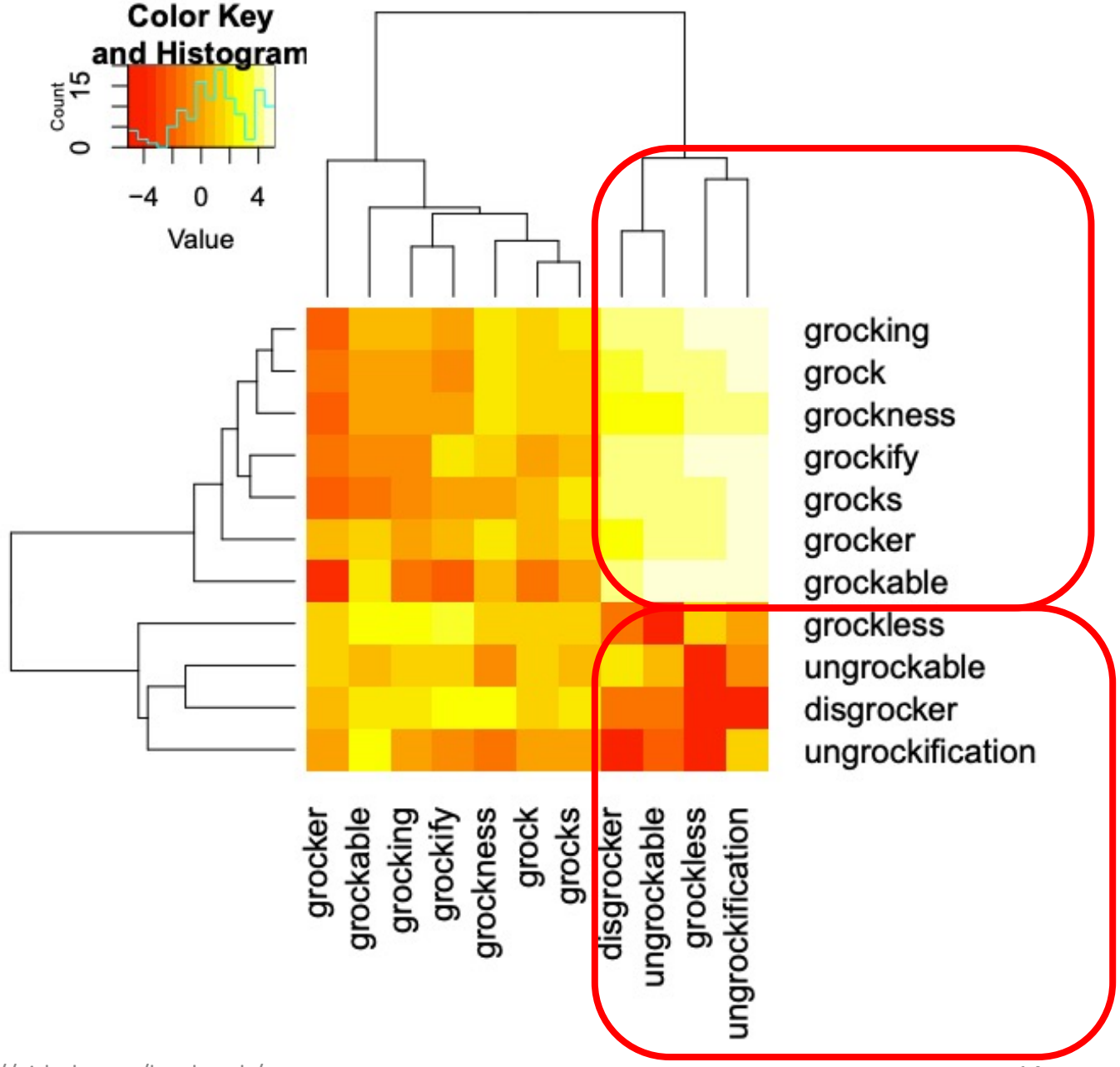
	black	ter	bad	evil	good	ff	white
black	-3.493	-5.090	-5.05	-5.05	3.59	-3.117	-4.46
ter	-5.107	-4.635	-5.06	-5.07	4.27	-0.517	3.99
bad	-5.051	-5.105	-5.06	-5.00	4.02	1.804	3.74
evil	-5.008	-5.042	-4.99	-4.99	4.35	2.685	4.28
good	4.297	4.098	4.54	4.49	-5.04	-5.127	-5.12
ff	-1.512	0.687	2.19	3.30	-2.56	-2.713	-3.16
white	-0.612	4.313	4.20	4.51	-3.52	-5.122	-5.07
	black	ter	bad	evil	good	ff	white
black	1.000	0.884	0.885	0.859	-0.918	-0.805	-0.772
ter	0.884	1.000	0.992	0.982	-0.981	-0.813	-0.743
bad	0.885	0.992	1.000	0.997	-0.995	-0.799	-0.753
evil	0.859	0.982	0.997	1.000	-0.991	-0.786	-0.749
good	-0.918	-0.981	-0.995	-0.991	1.000	0.819	0.784
ff	-0.805	-0.813	-0.799	-0.786	0.819	1.000	0.938
white	-0.772	-0.743	-0.753	-0.749	0.784	0.938	1.000

Table 8: Biases in output model. Top (logits); Bottom (correlations of logits). Positive logits → antonyms. Headings are abbreviations for words in Figure 1.

Clustering of Cor of Logits

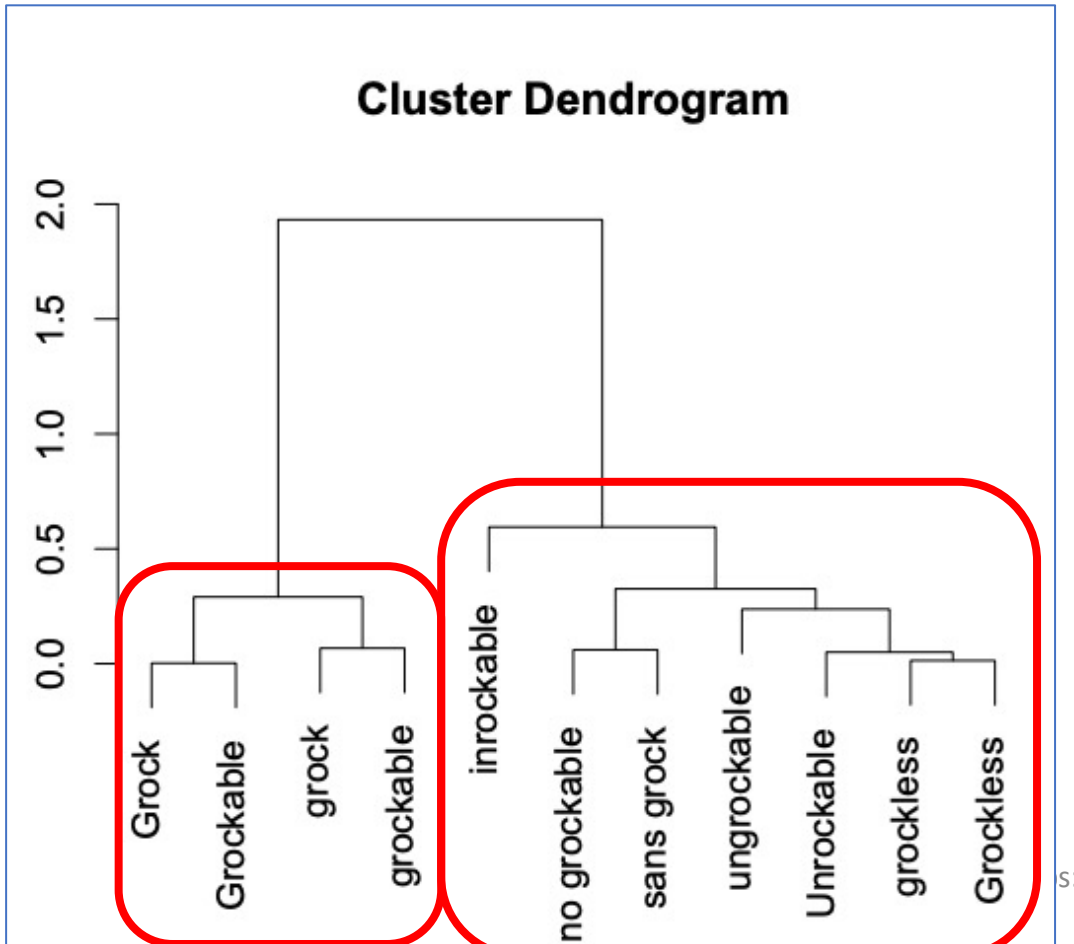


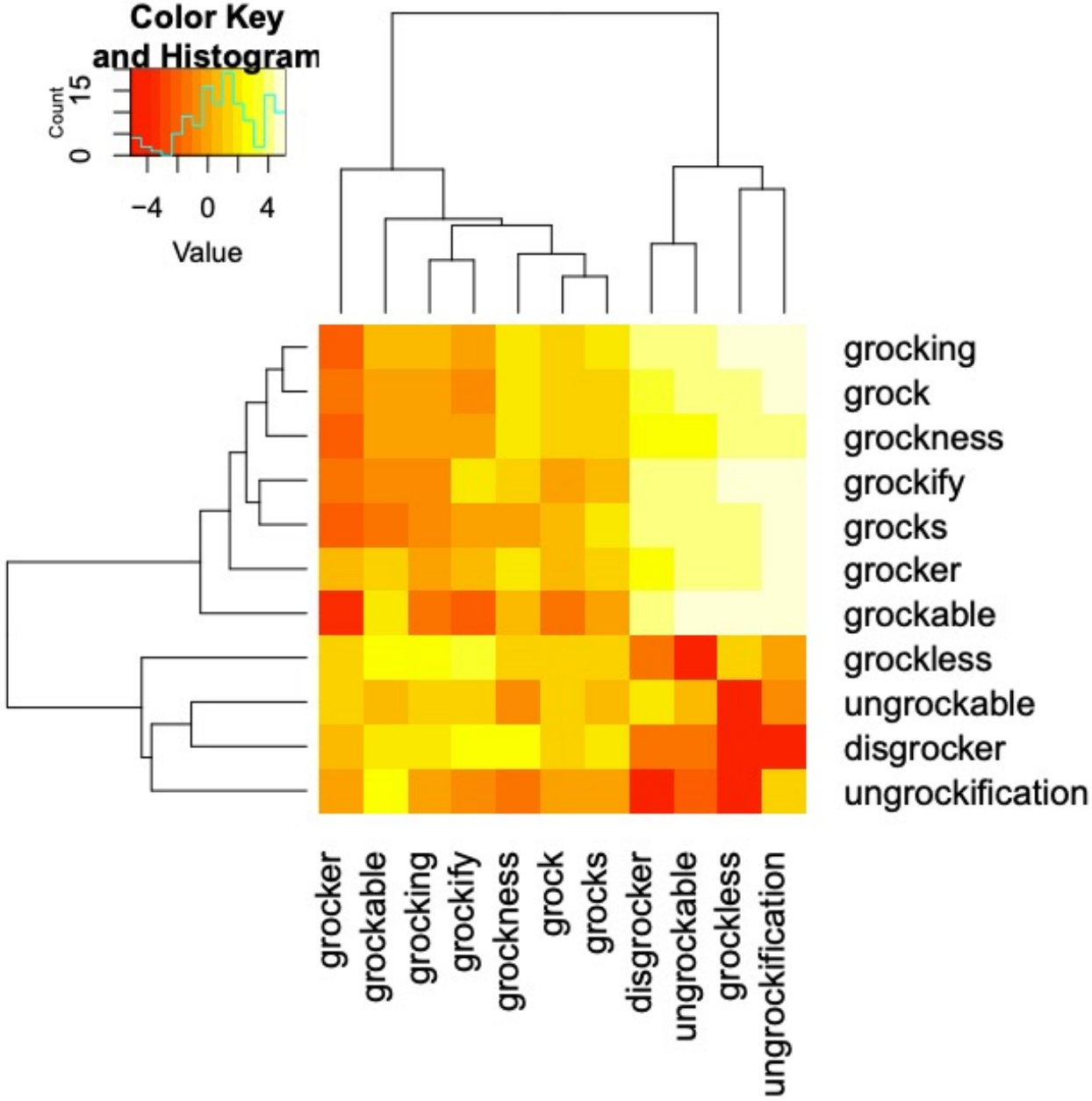
From Words to Text
(OOVs: Out of Vocabulary)



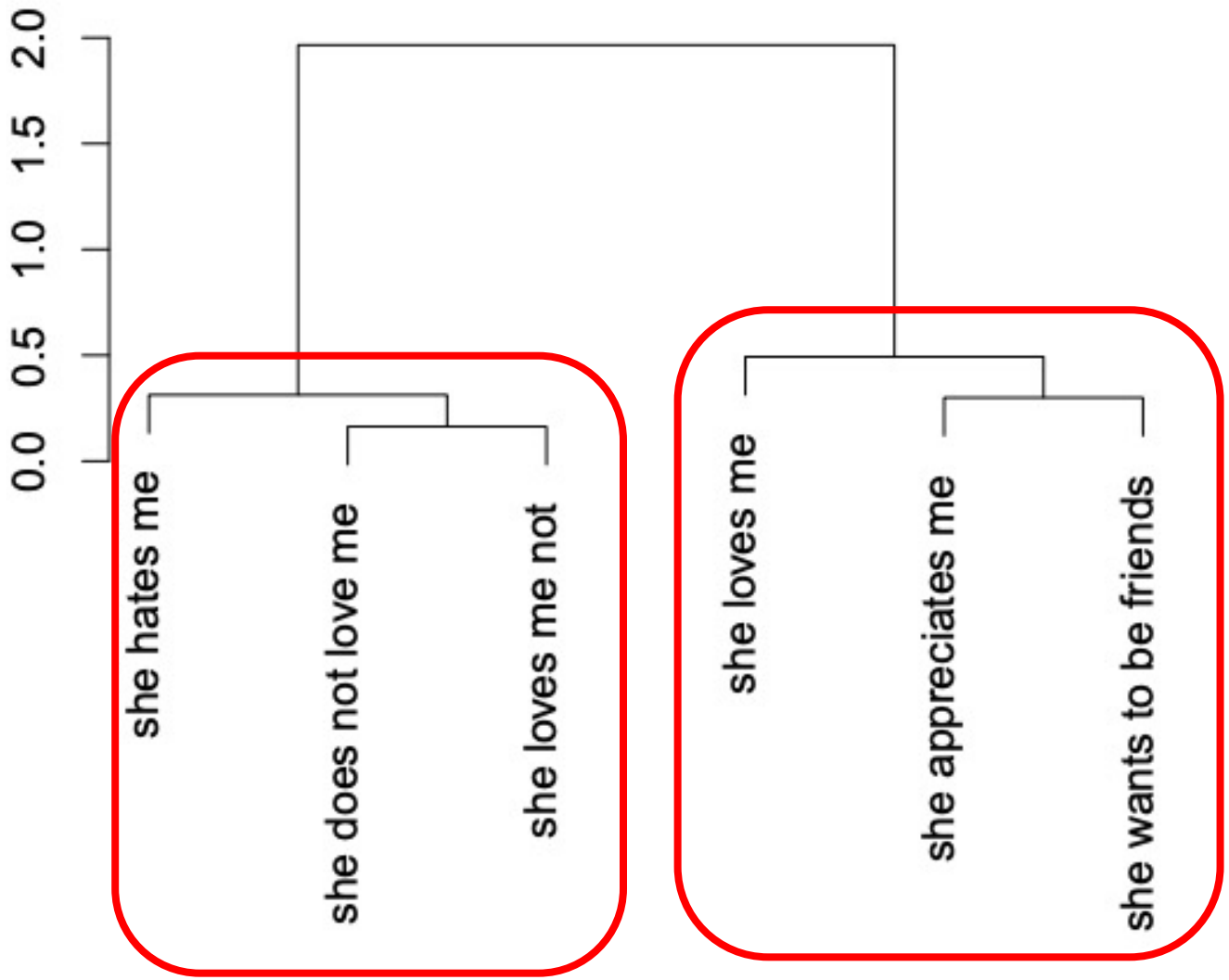
From Words to Text
(OOVs: Out of Vocabulary)

Google Translations of Variants of *Grock*





From Words to Text (Negation)



Agenda

- ✓Syn/Ant Binary Classification
- ✓From Words to Texts
  - ✓MWEs: Multiword Expressions
  - ✓OOVs: Out of Vocabulary words
  - ✓Multi-Lingual
  - ✓Negation
- ➢**Leakage with Standard Benchmarks**
- VAD Regression
  - VAD = Valance, Arousal, Dominance

Concerns about Leakage

Most Graphs are Sparse

training set	V	E	CC
adj	3315	5562	285
noun	3654	2836	1204
verb	1859	2534	199
fallows	15,466	58,494	32
fallows-s	6326	5886	907
SimLex	1028	999	151
NRC-VAD	20,007	$20,007^2$	1

Table 9: Most graphs are sparse, $E \ll V^2$, except NRC-VAD. V (vertices), E (edges) and CC (connected components) are computed over training sets.

Concerns

- Clusters (above) suggest structures
  - (Too) Simple Approximations
    - Equivalence Relations (Synonyms)
      - Symmetry
        - $a = b \Rightarrow b = a$
      - Transitivity
        - $a = b \,\&\, b = c \Rightarrow a = c$
    - Anti-symmetry (Antonyms)
      - $a \neq b \Rightarrow b \neq a$
  - Lots of Connected Components
- Risk of Leakage
  - Are there clues for connecting components across splits (train, val, test)?

Evidence for Leakage

Paths of Length 1

- Consider 99 edges of length 1
  - Example: *good* ↔ *awful*
- These are particularly worrisome.
  - The same edge is in
  - both train and validation splits,
  - but in different directions
- These 99 pairs are clearly leaking information across splits

Many Short Paths

Path Length	adj	noun	verb	fallows
0				2
1	99	59	60	946
2	80	7	15	3835
3	59	3	7	1156
4+	70	2	35	639
NA	90	135	65	612
total	398	206	182	7190

Table 10: For most pairs of words in the validation set, w_1 and w_2, there is a short path from w_1 to w_2 based on edges in the training set.

Evidence for Leakage

Paths of Length 2

- Paths of length 2
  - Not at bad as paths of length 1
  - but concerning, nevertheless
- Examples:
  - *innocent* ↔ *harmful* ↔ *harmless*
  - *fresh* ↔ *aged* ↔ *old*
  - *dead* ↔ *alive* ↔ *deceased*
- How can we use these paths to leak labels across splits?

Many Short Paths

Path Length	adj	noun	verb	fallows
0				2
1	99	59	60	946
2	80	7	15	3835
3	59	3	7	1156
4+	70	2	35	639
NA	90	135	65	612
total	398	206	182	7190

Table 10: For most pairs of words in the validation set, w_1 and w_2, there is a short path from w_1 to w_2 based on edges in the training set.

A-Leakage

- Definitions
  - Let $e = (a, b)$ be an edge in val split
  - Let $label_v(e)$ be the label on e in val
  - Let $path_t(a, b)$ be the shortest path
    - from a to b using edges from train
  - Let A_t be the number of antonym labels on $path_t(e)$
- A-Leakage Heuristic:
  - $label_v(e) \approx$ antonym iff A_t is odd

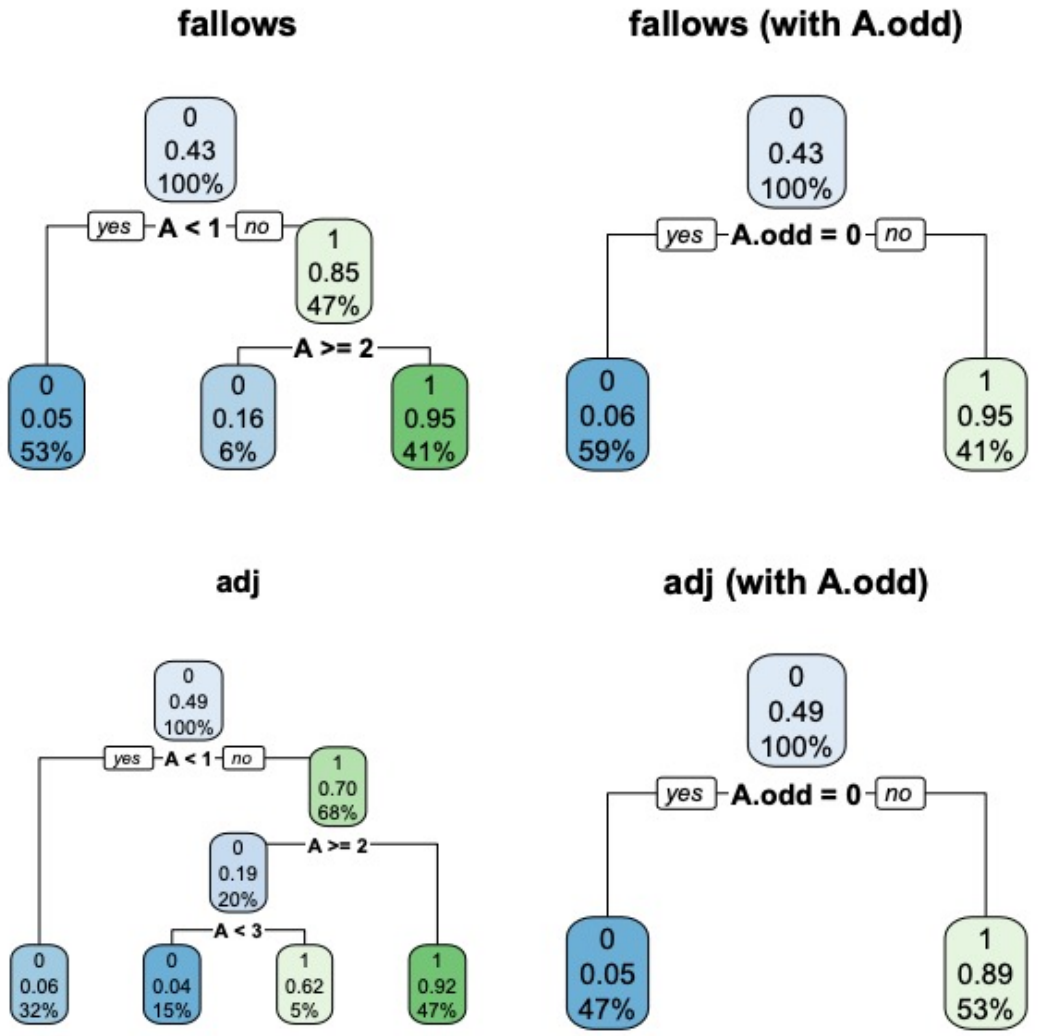


Figure 5: *A-leakage*: Decision trees learn to classify pairs as antonyms iff A is odd.

A-Leakage is Serious

Bad Benchmarks → Retractions

	val		test	
	acc	applicable	acc	applicable
adj	0.916	308/398	0.906	1482/1986
noun	0.930	72/206	0.983	302/1020
verb	0.872	118/182	0.882	587/908
fallows	0.945	6576/7190	0.949	6722/7366
fallows-s	0.683	223/753	0.694	241/777

Table 11: A-leakage: $Pr(ant) > Pr(syn)$ iff A is odd. Accuracy is computed over applicable edges. Denominators are borrowed from Table 3.

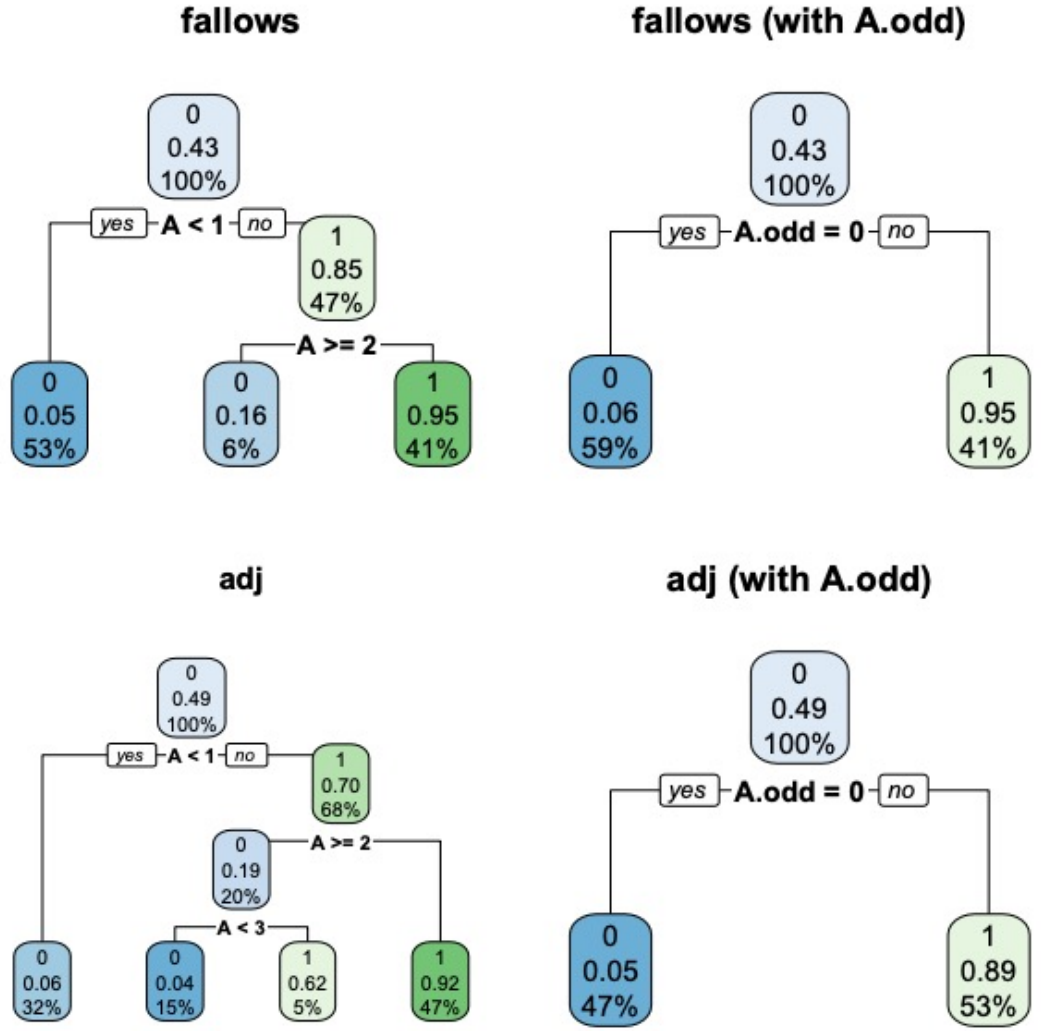


Figure 5: *A-leakage*: Decision trees learn to classify pairs as antonyms iff A is odd.

So Many Bad Benchmarks… So Few Retractions

Church & Bian (EMNLP-2021)

WN18 → WN18RR

Relation	Edges	Inverse	Edges
hypernyms	37,221	hyponyms	37,221
derivationally related forms	31,867		
member meronym	7928	member holonum	7928
has part	5142	part of	5148
synset domain topic of	3335	member of domain topic	3341
instance hypernym	3150	instance hyponym	3150
also see	1396		
verb group	1220		
member of domain region	983	synset domain region of	982
member of domain usage	675	synset domain usage of	669
similar to	86		

Table 2: 18 Relations in WN18. By construction, many of these relations have inverses (with similar counts).

WN18RR

Forward Links	Inverse Links: yRx test	train	valid	Totals
xRy **test**	24	1011	39	1074
xRy **train**	1011	27,701	1003	29,715
xRy **valid**	39	1003	36	1078
Totals	1074	29,715	1078	31,867

Table 4: Information Leakage in WN18RR: Derivationally related links are symmetric ($xRy \Rightarrow yRx$).

Agenda

- ✓Syn/Ant Binary Classification
- ✓From Words to Texts
  - ✓MWEs: Multiword Expressions
  - ✓OOVs: Out of Vocabulary words
  - ✓Multi-Lingual
  - ✓Negation
- ✓Leakage with Standard Benchmarks
- ➢**VAD Regression**
  - VAD = Valance, Arousal, Dominance

Another Example of a Lexical Resource: VAD https://saifmohammad.com/WebPages/nrc-vad.html

- History in Psycholinguistics: (Osgood et al., 1957; Russell, 1980, 2003)
  - Valence: *positive-negative* or *pleasure-displeasure*
  - Arousal: *excited-calm* or *active--passive*
  - Dominance: *powerful-weak* or *have full control-have no control*

Entries with Highest and Lowest Scores in the VAD Lexicon

Dimension	Word	Score↑	Word	Score↓
valence	*love*	1.000	*toxic*	0.008
	happy	1.000	*nightmare*	0.005
	happily	1.000	*shit*	0.000
arousal	*abduction*	0.990	*mellow*	0.069
	exorcism	0.980	*siesta*	0.046
	homicide	0.973	*napping*	0.046
dominance	*powerful*	0.991	*empty*	0.081
	leadership	0.983	*frail*	0.069
	success	0.981	*weak*	0.045

Average Split-Half Reliability Scores for VAD Annotations:
higher scores indicate higher reliability

Annotations	# Terms	V	A	D
NRC VAD Lexicon	20,007	0.95	0.90	0.90
NRC VAD Lexicon (terms common to Warriner lexicon)	13,915	0.95	0.91	0.91
Warriner et al. Lexicon	13,915	0.91	0.79	0.77

Syn/Ant Classification → VAD Regression

- Motivation: classify → regress
  - Concerns about leakage
- NRC-VAD is similar to sym/lex
  - but lexicon is fully-connected
- 20k lemmas, w, where $VAD(w) \in \mathbb{R}^3$
  - $y(w_1, w_2) = |VAD(w_1) - VAD(w_2)|$
- Regression: $y \sim w_1 + w_2$
- Standard test/val/train splits:
  - Split lexicon by E
- But for generalizations to OOVs
  - Might be more interested in splits by V (w)

word	Val	Arousal	Dom	Dist
open	0.620	0.480	0.569	0.00
unfold	0.612	0.510	0.520	0.06
reopen	0.656	0.528	0.568	0.06
close	0.292	0.260	0.263	0.50
closed	0.240	0.164	0.318	0.55
undecided	0.286	0.433	0.127	0.56

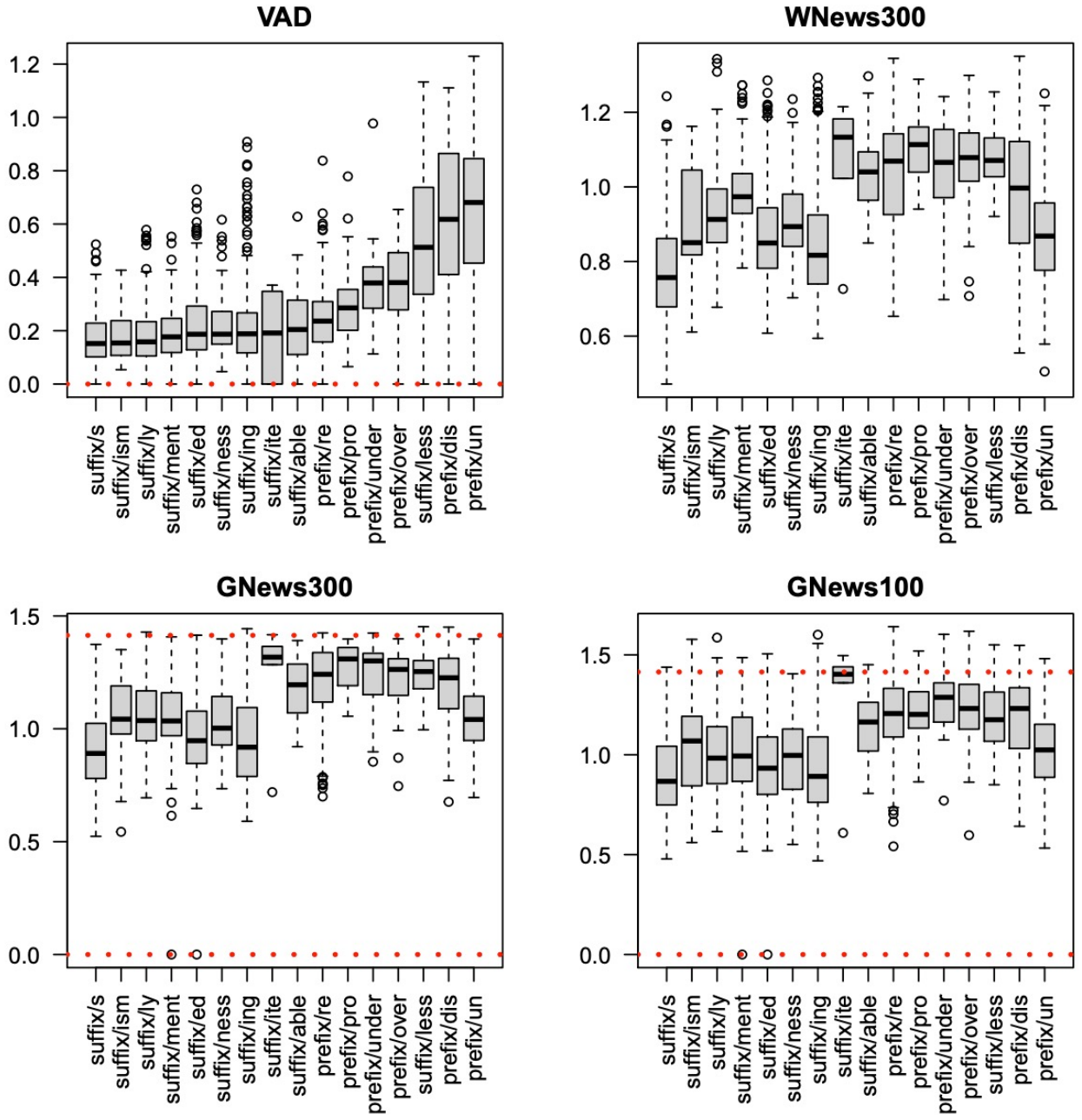
Table 12: Words above the double line are near *open*. The last column is the Euclidean distance to *open*.

	Antonyms				Sim
	adj	noun	verb	fallows	SimLex
cor	0.55	0.48	0.44	0.52	-0.40

Table 13: VAD distances are positively correlated with antonyms, and negatively correlated with SimLex similarities, though none of these correlations are large.

Morpheme Diagnostic

- Group words by affixes
  - *over-*
    - *overtake/take*
    - *overlook/look*
- Plot y for pairs in each group
  - $y(w_1, w_2) = |VAD(w_1) - VAD(w_2)|$
- **Red** baselines:
  - 0: distance for maximally similar pair
  - $\sqrt{2}$: distance for random pair
- Observations:
  - VAD varies systematically:
    - Small (similar in VAD): *-s, -ism, -ly*
    - Large (dissimilar in VAD): *-less, dis-, un-*
  - Word2vec is large (almost everywhere)
    - Almost all pairs of words are far apart
    - Even words that are morphologically related



VAD Results (R2)
R2 →1.0 (good); R2 → 0 (bad)

- Train/Val/Test splits
  - Based on $V = 16k$ (of $20k$)
  - Remainder held-out to test generalizations to OOVs
- Results are promising when
  - Splits are large and
  - Representative of one another
- Experimented with training sets of 10, 100k and 1M edges

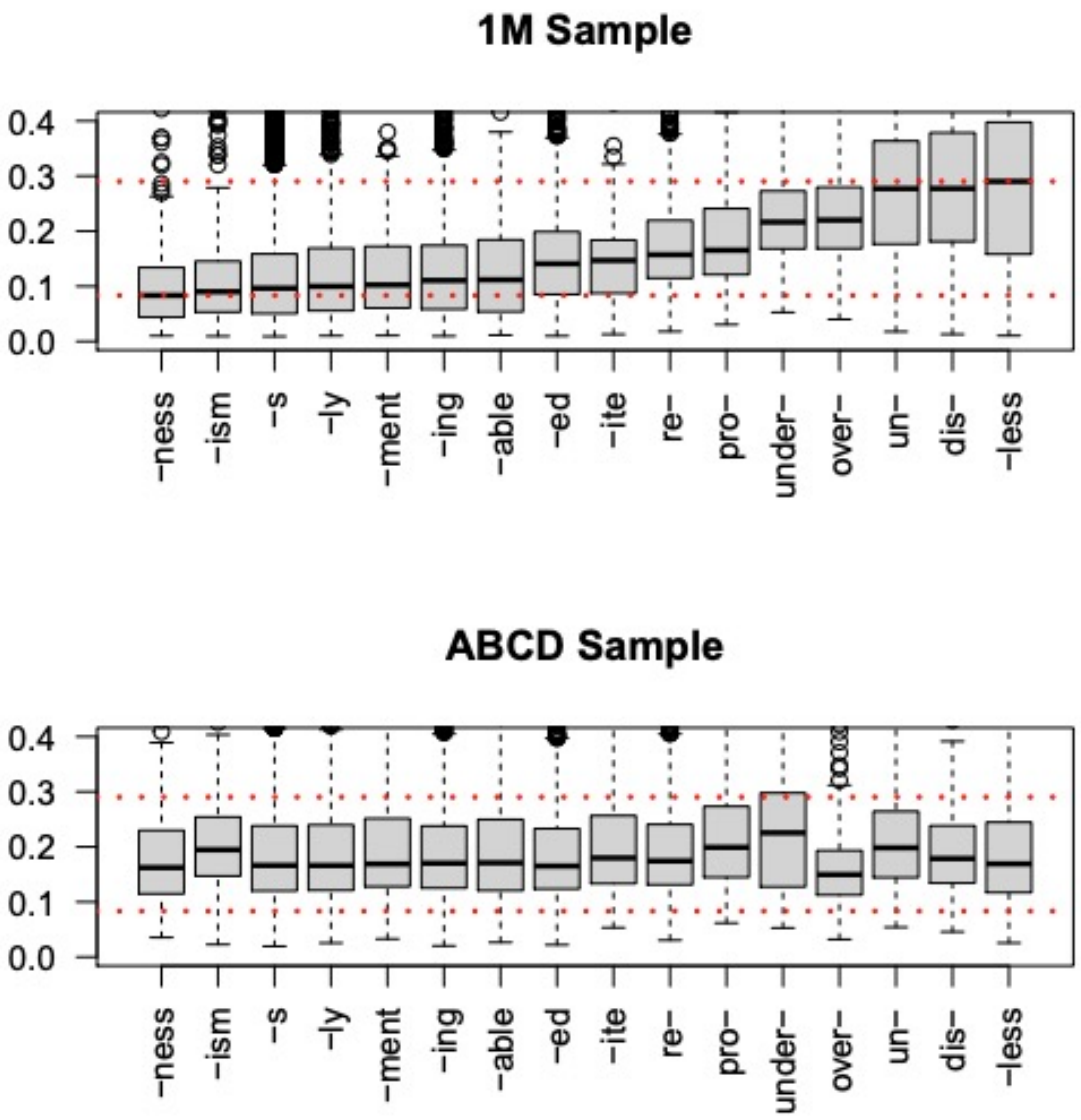
Promising Transfer: Train with 1M Edges

$R2(test) \approx R2(val) \approx R2(train) \approx 1$

base model	train	val	test
BERTun	0.993	0.993	0.993
SciBERTun	0.993	0.993	0.992
ERNIE	0.991	0.990	0.990
SciBERTc	0.988	0.988	0.987
BERTmulti	0.988	0.987	0.991
BERTc	0.995	0.995	0.988

Morpheme Diagnostic (with ABCD Sampling)

- ABCD Sampling:
  - Split V in A, B, C, D randomly
  - Sample edges such that
    - Training Split: $a \rightarrow b$
    - Validation Split: $a \rightarrow c$
    - Test Split: $a \rightarrow d$
  - where $a \in A, b \in B, c \in C, d \in D$
- Unfortunately,
  - ABCD sampling does not transfer well,
  - and does not pass morpheme diag



Agenda

- ✓Syn/Ant Binary Classification
- ✓From Words to Texts
  - ✓MWEs: Multiword Expressions
  - ✓OOVs: Out of Vocabulary words
  - ✓Multi-Lingual
  - ✓Negation
- ✓Leakage with Standard Benchmarks
- ✓VAD Regression
  - ✓VAD = Valance, Arousal, Dominance

Conclusions

- Proposed fine-tuning deep nets on lexical resources
  - Thesaurus (syn/ant classification)
  - VAD Regression
  - $y \sim text_1 + text_2$
- Proposed method is competitive with MoE baseline, and
  - Generalizes better to Fallows (1898)
- Words → Texts
  - Proposed method can be applied at inference time to MWEs, OOVs and longer tests in multiple languages
- On a cautionary note,
  - found evidence of leakage
    - in standard benchmarks as well as Fallows (1898)
  - Work based on bad benchmarks
    - may need to be retracted
- To address concerns with leakage,
  - we introduced a new task: VAD regression
  - Since VAD is fully-connected,
    - we could study sampling methods
- Transfer is more effective
  - when splits are large
    - and representative of one another
  - In such cases,
    - reduces training loss (in fine-tuning)
    - also reduces loss on other splits
- Proposed method:
  - effective for pairs of words in training set
  - but less so for pairs of unseen words