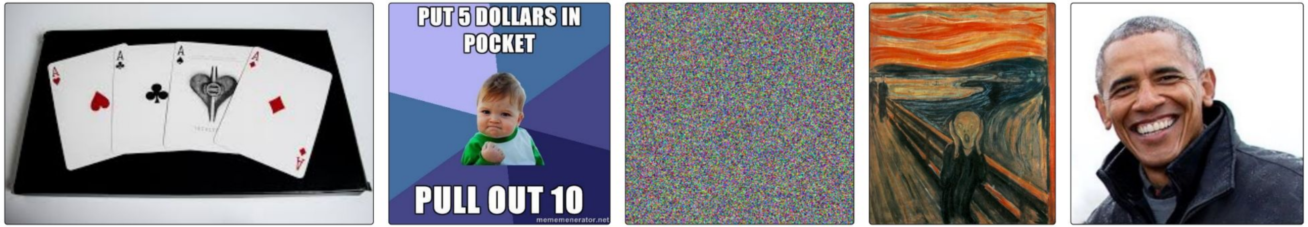
Out-of-distribution detection for neural NLP models

Hrant Khachatrian @ YerevaNN

Joint work with Karen Hambardzumyan

#AMLD2019

Motivation: image classification



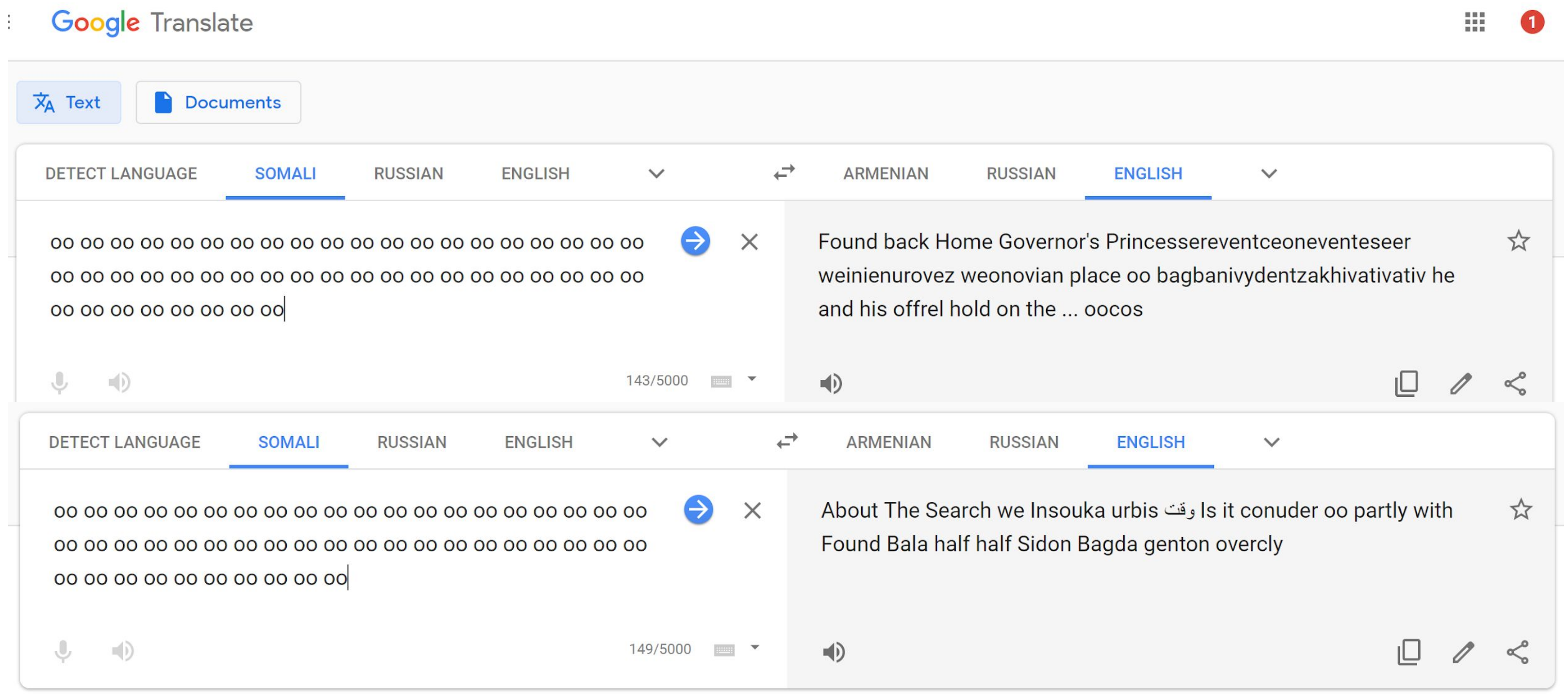
DenseNet 161 (2017)	Envelope 31%	Balance Beam 52%	Chainlink Fence 31%	Chest 37%	Tench 36%
SqueezeNet (2016)	Binder 43%	Balance Beam 18%	Poncho 32%	Jean 30%	Suit 21%
ResNet 152 (2015)	Envelope 40%	Pacifier 33%	Chain Mail 29%	Dust Cover 52%	Sweatshirt 25%
VGG 19 (2014)	Binder 51%	Dust Cover 44%	Window Screen 5%	Chest 11%	Sweatshirt 46%
AlexNet (2012)	T-shirt 16%	Dust Cover 22%	Cardigan 12%	Theater Curtain 3%	Coho 37%

Figure 1: The arbitrary predictions of several popular networks [2, 3, 4, 5, 6] that are trained on ImageNet [1] on unseen data. The red predictions are entirely wrong, the green predictions are justifiable, the orange predictions are less justifiable. The middle image is noise sampled from $\mathcal{N}(\mu = 0.5, \sigma = 0.25)$ without any modifications. This unpredictable behaviour is not limited to demonstrated architectures. We show that merely thresholding the output probability is not a reliable method to detect these problematic instances.

Figure taken from [Shafaei et al., 2018]

Alireza Shafaei, Mark Schmidt, and James J. Little, *Does Your Model Know the Digit 6 Is Not a Cat? A Less Biased Evaluation of "Outlier" Detectors*

Motivation: machine translation



oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo

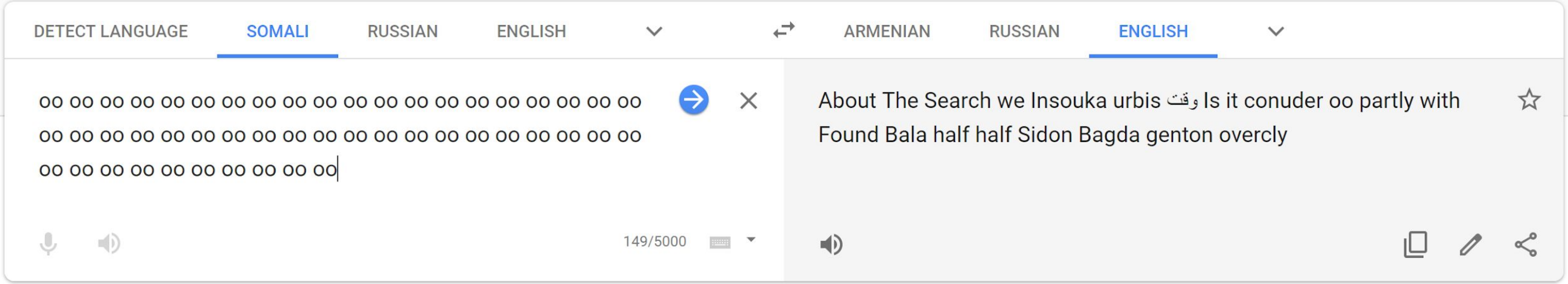
Found back Home Governor's Princessereventceoneventeseer weinienurovez weonovian place oo bagbanivydentzakhivativ he and his offrel hold on the ... oocos

oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo

About The Search we Insouka urbis وقت Is it conuder oo partly with Found Bala half half Sidon Bagda genton overcly

The problem

- Can we look at the input sample and the output of the neural network and figure out whether we should trust the output?
  - Intuitively, “oo oo oo ….” is very far from the all the sentences used in the training process. Therefore, we should not expect reasonable output from the network.
  - Can we “quantify” this?



oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo oo

About The Search we Insouka urbis وقت Is it conuder oo partly with Found Bala half half Sidon Bagda genton overcly

Basic approach

- [Hendrycks and Gimpel, ICLR 2017] proposed a simple baseline
- *“Correctly classified examples tend to have greater **maximum softmax probabilities** than erroneously classified and out-of-distribution examples, allowing for their detection”*
- Let f(x) be the output of the last layer of the neural network (before softmax)

$$score_{base} = \max softmax(f(x))$$

- Use this score to discriminate correctly and incorrectly classified examples:

  AUC = 0.93 for CIFAR-10 test set

  AUC = 0.87 for CIFAR-100 test set

Basic approach

- [Hendrycks and Gimpel, ICLR 2017] proposed a simple baseline
- *"Correctly classified examples tend to have greater **maximum softmax probabilities** than erroneously classified and out-of-distribution examples, allowing for their detection"*
- Let f(x) be the output of the last layer of the neural network (before softmax)

$$score_{base} = \max softmax(f(x))$$

- Use this score to discriminate CIFAR-10 and LSUN test set samples:

  AUC = 0.95 for DenseNet-101

Better approach: ODIN

- [Liang et al., ICLR 2018] proposed an improvement
- ODIN algorithm adds two tricks:
  - Use adversarial-like perturbation
  - Use high temperature softmax

$$\hat{x} = x + \epsilon \operatorname{sgn}(\nabla_x f_{\hat{y}}(x))$$

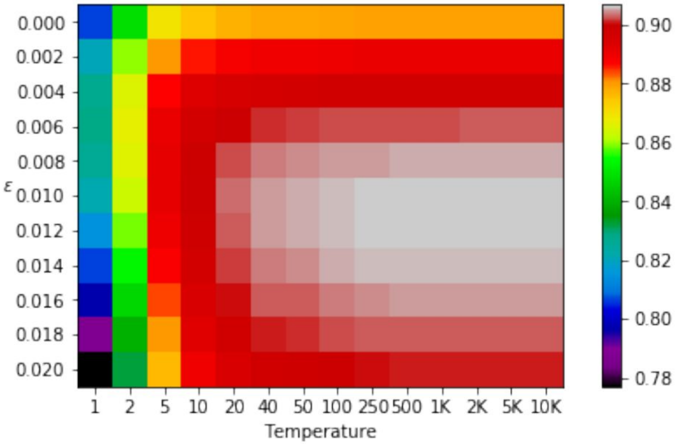
$$score_{\text{ODIN}} = \max softmax_T\left(f\left(\hat{x}\right)\right)$$

- Use this score to discriminate CIFAR-10 and LSUN test set samples:

  AUC = 0.98 for DenseNet-101 (vs 0.95 of the baseline)

Do these techniques work for NLP tasks?

- Sentiment analysis
  - A simple bi-LSTM trained on Yelp Reviews dataset
  - Discriminate sentences from Yelp Reviews and Stanford Sentiment Treebank
  - AUC = 0.907



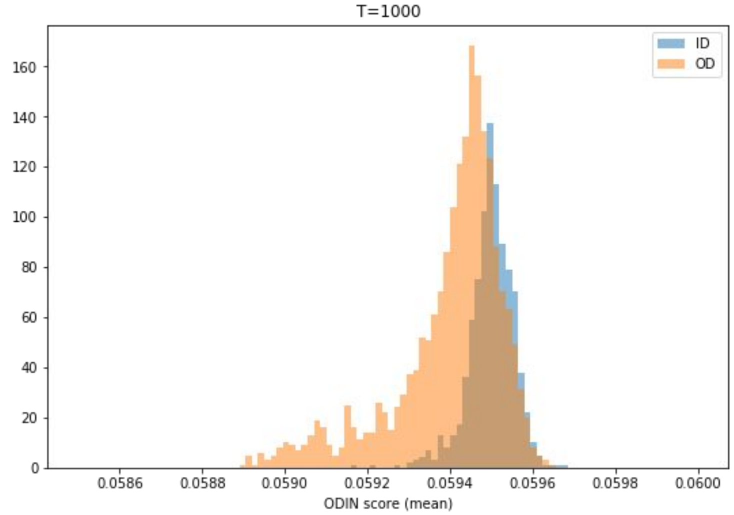
There is an optimal value for ε

AUC

High temperatures are better

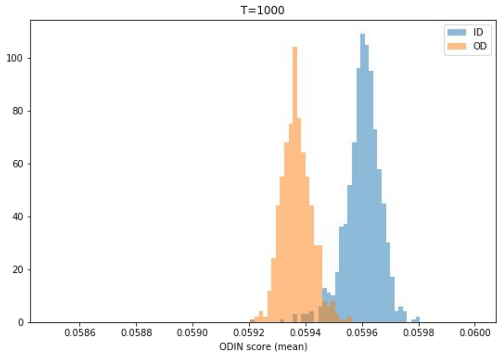
Do these techniques work for NLP tasks?

- Part-of-speech tagging
  - We train on UD English-LinES
  - We test on two datasets:
    - **UD English-EWT: AUC=0.751**
      - Probably because EWT has a subset very similar to LinES

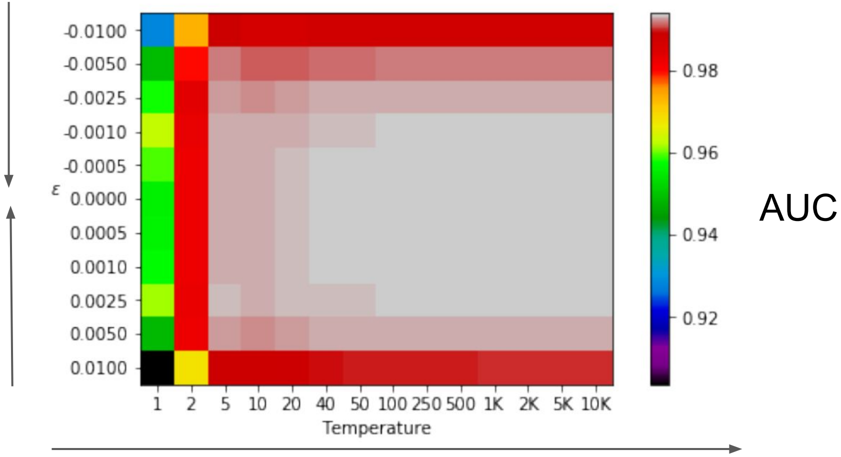


Do these techniques work for NLP tasks?

- Part-of-speech tagging
  - We train on UD English-LinES
  - We test on two datasets:
    - UD English-EWT: AUC=0.751
      - Probably because EWT has a subset very similar to LinES
    - **UD Dutch-Alpino: AUC=0.991**
      - ε>0 doesn’t help!



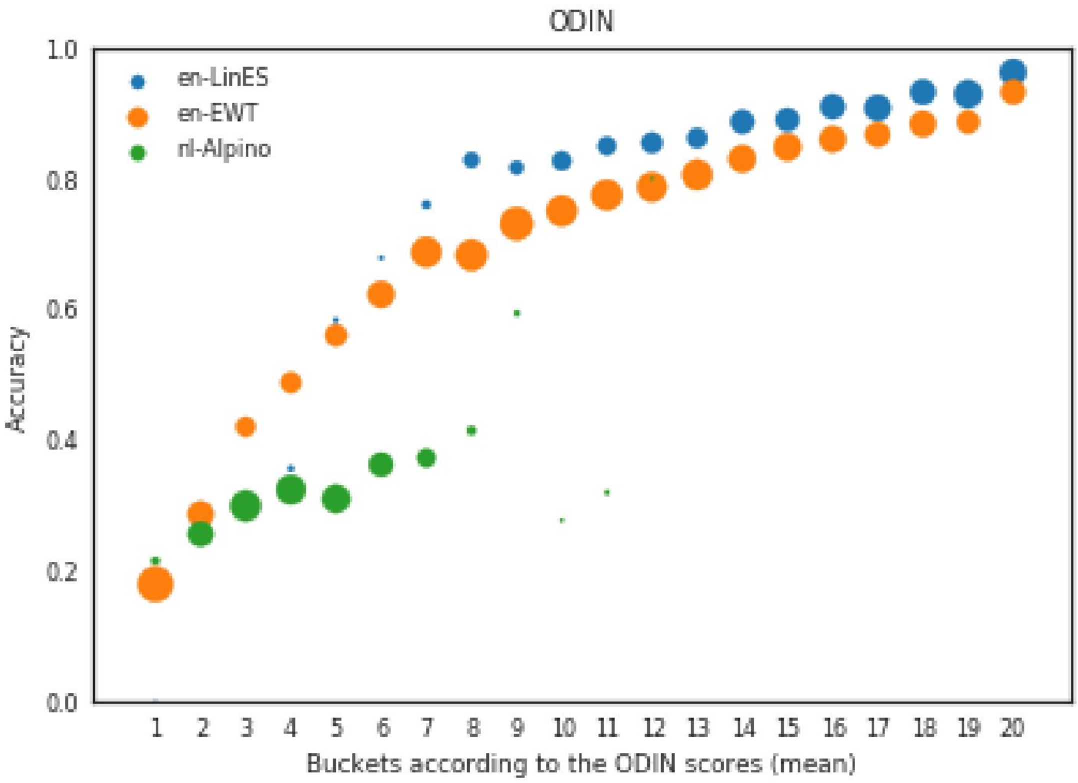
ε=0 is the best one



High temperatures are still better

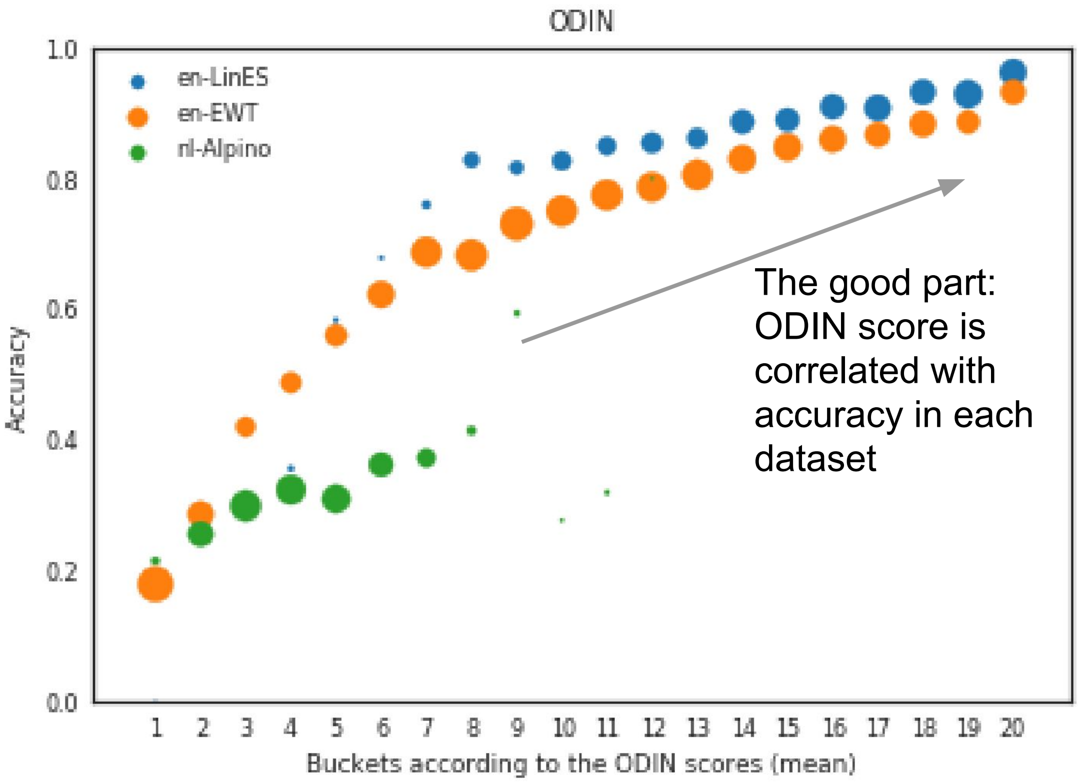
What does it mean for practice?

- Train on English LinES
- Combine the samples from the test sets of 3 datasets
- Order the samples by ODIN score and split them into 20 buckets
- Calculate POS tagging accuracy for each (bucket, dataset) pair



What does it mean for practice?

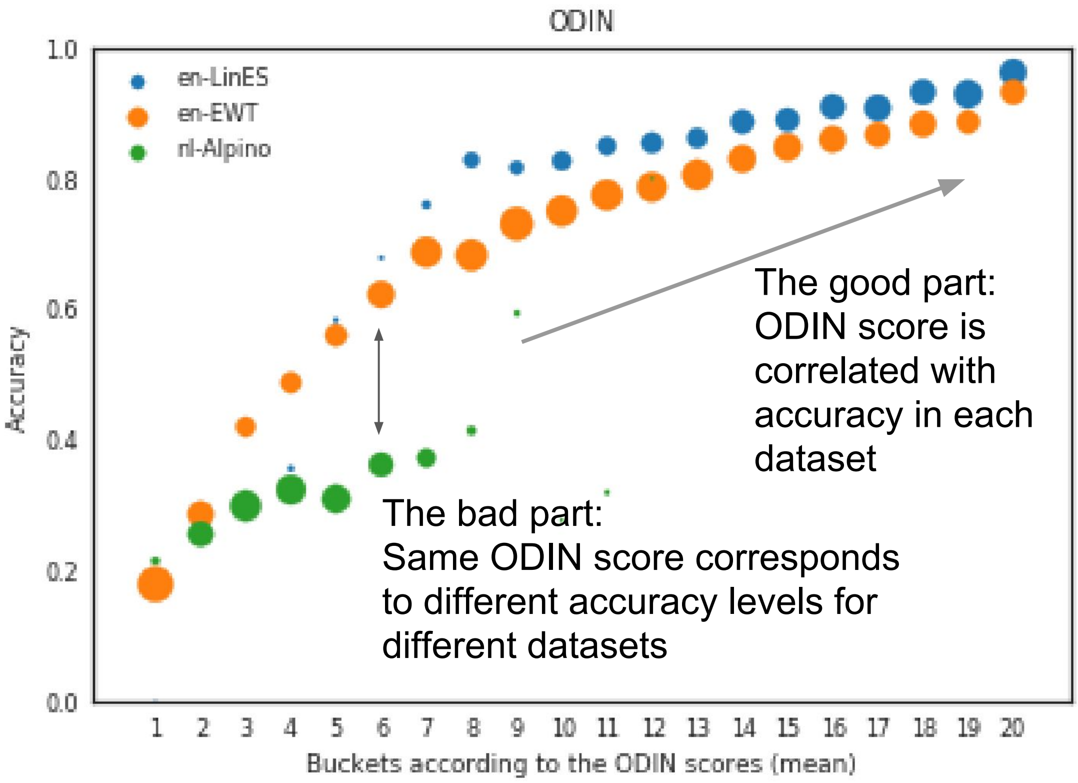
- Train on English LinES
- Combine the samples from the test sets of 3 datasets
- Order the samples by ODIN score and split them into 20 buckets
- Calculate POS tagging accuracy for each (bucket, dataset) pair



The good part: ODIN score is correlated with accuracy in each dataset

What does it mean for practice?

- Train on English LinES
- Combine the samples from the test sets of 3 datasets
- Order the samples by ODIN score and split them into 20 buckets
- Calculate POS tagging accuracy for each (bucket, dataset) pair



The good part:
ODIN score is correlated with accuracy in each dataset

The bad part:
Same ODIN score corresponds to different accuracy levels for different datasets

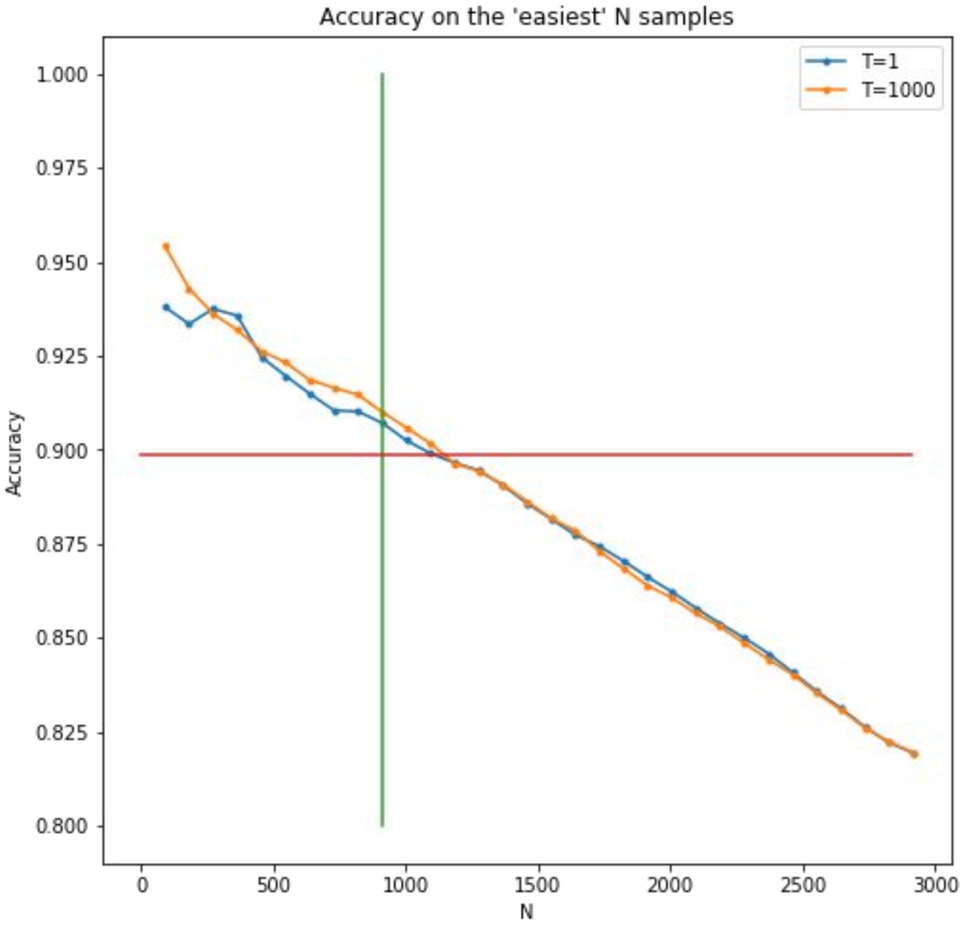
Thanks

https://yerevann.com

YN^2

Part-of-speech tagging

ODIN score can be used as a confidence measure on both datasets.


Accuracy on the 'easiest' N samples
T=1
T=1000
Accuracy
N