University of Massachusetts
UMASS Medical School
The academic partner of UMass Memorial Health Care

Machine Learning Intelligence

Hong Yu, and the UMass BioNLP group

University of Massachusetts Medical School
University of Massachusetts-Amherst
Bedford VAMC

NLP Challenges for Detecting Medication and Adverse Drug Events from Electronic Health Records (MADE1.0)

hosted by University of Massachusetts Medical School

Adverse drug events (ADEs) are common and occur in approximately 2-5% of hospitalized adult patients. Each ADE is estimated to increase healthcare cost by more than $3,200. Severe ADEs rank among the top 5 or 6 leading causes of death in the United States. Prevention, early detection and mitigation of ADEs could save both lives and dollars. Employing natural language processing (NLP) techniques on electronic health records (EHRs) provides an effective way of real-time pharmacovigilance and drug safety surveillance.

We've annotated 1092 EHR notes with medications, as well as relations to their corresponding attributes, indications and adverse events. It provides valuable resources to develop NLP systems to automatically detect those clinically important entities. Therefore we are happy to announce a public NLP challenge, MADE1.0, aiming to promote deep innovations in related research tasks, and bring researchers and professionals together exchanging research ideas and sharing expertise. The ultimate goal is to further advance ADE detection techniques to improve patient safety and health care quality.

Tentative Timelines

- Registration: begins August 1st, 2017
- Training data release: October 2nd, 2017
- System submission: Jan 2nd, 2018
- Workshop: in conjunction with AMIA summit 2018, March 2018

Annotated Data

The entire dataset contains 1092 de-identified EHR notes from 21 cancer patients. Each EHR note was annotated with medication information (medication name, dosage, route, frequency, duration), ADEs, indications, other signs and symptoms, and relations among those entities. We split the data into a training set consisting of ~900 notes and a test set consisting of ~180 notes. Both will be released in BioC format.

Data Statistics

Labels	Annotations	Avg. Annotation Length
ADE	905	1.51
Indication	1988	2.34
Other SSD	26013	2.14
Severity	1928	1.38
Drugname	9917	1.20
Duration	562	2.17
Dosage	3284	2.14
Route	1810	1.14
Frequency	2801	2.35

Duplication: A challenge for Data Mining

- Previously showed that 40% EHR content was duplicated
- We define three types of duplications
  - Exact copy and paste
  - Approximate copy and paste
  - Event repeat
- Our findings:
  - 23% events were duplicated
  - However, only 6% ADEs were duplicated

Jagannatha and Yu. 2017. Duplication of data in clinical narratives in electronic health records. AMIA submission 2017.

Challenges for ADE Detection

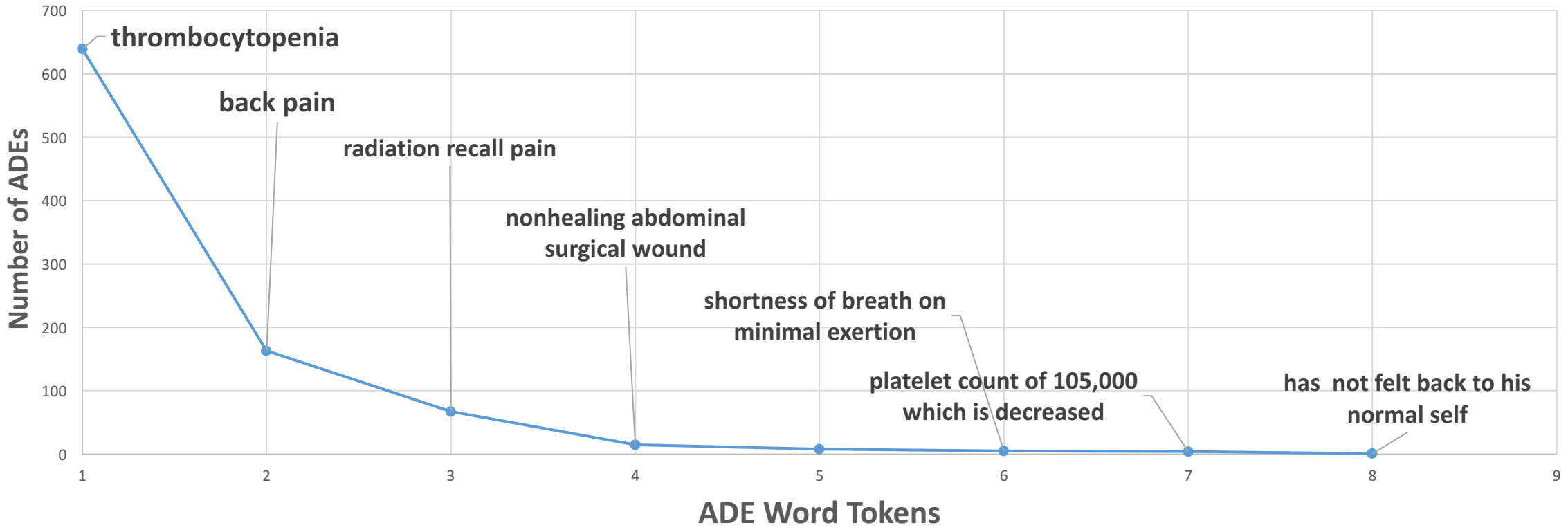
- Common vocabulary between different medical entities. E.g.

Example Sentence from Dataset

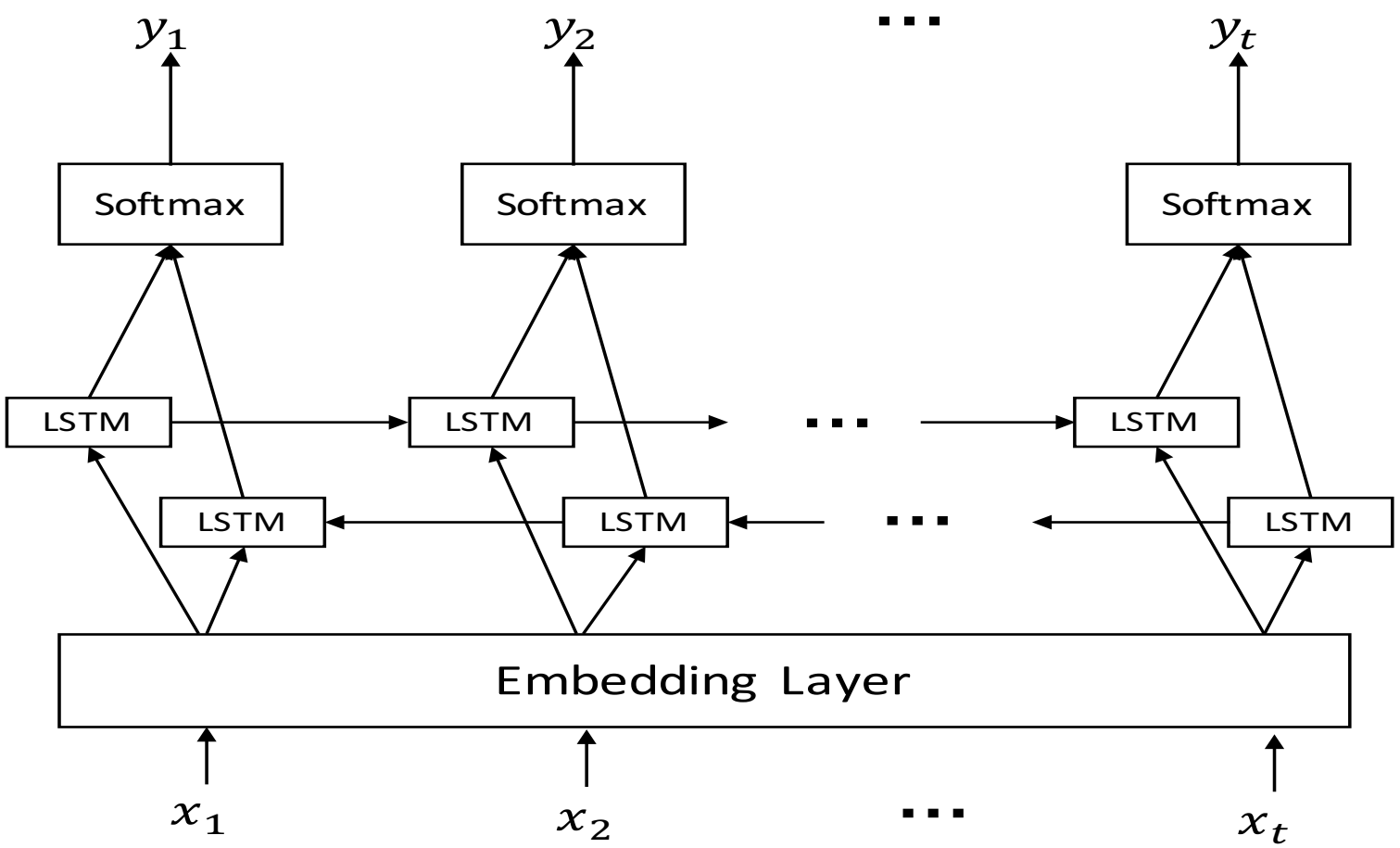
Anemia and *Thrombocytopenia*: Anemia likely multifactorial d/t CTCL, chronic disease, and romidepsin; *thrombocytopenia* likely d/t romidepsin. -Discussed with Dr. [**Last Name (STitle) 33**]. *Thrombocytopenia* will not likely worsen much further. [**Doctor First Name **] to continue coumadin for now.

  - In our corpus, an Adverse Drug Event is tagged only when there is a direct evidence in the text , linking it as a side effect of a medication.
  - The second mention of *thrombocytopenia* is an Adverse Drug Event.
  - The first and third mentions of *Thrombocytopenia* are not used to describe a side effect directly, so they are labeled as *Other SSD* .

Challenges


700
600
500
400
300
200
100
0
Number of ADEs
thrombocytopenia
back pain
radiation recall pain
nonhealing abdominal surgical wound
shortness of breath on minimal exertion
platelet count of 105,000 which is decreased
has not felt back to his normal self
1
2
3
4
5
6
7
8
9
ADE Word Tokens

Bidirectional RNNs



Jagannatha and Yu. 2016. Bidirectional recurrent neural networks for medical event detection in Electronic Health Records. NAACL 2016.

Results

Models	Recall	Precision	F-score
CRF-nocontext	0.6562	0.7330	0.6925
CRF-context	0.6806	0.7711	0.7230
LSTM-sentence	0.8024	0.7803	0.7912
GRU-sentence	0.8013	0.7802	0.7906
LSTM-document	0.8050	0.7796	0.7921
GRU-document	**0.8126**	**0.7938**	**0.8031**

Strict Evaluation results for micro averaged Recall, Precision and F-score. All results use ten fold cross validation.

Jagannatha and Yu. 2016. Bidirectional recurrent neural networks for medical event detection in Electronic Health Records. NAACL 2016.

LSTM-CRF

Models / Metrics	Strict Evaluation (Phrase Based)			Relaxed Evaluation (Word Based)		
	Recall	Precision	F-score	Recall	Precision	F-score
Bi-LSTM	0.8101	0.7845	0.7971	0.8402	0.8720	0.8558
Bi-LSTM-CRF	0.7890	0.8066	0.7977	0.8068	***0.8839***	0.8436
Bi-LSTM-CRF-pair	0.8073	***0.8266***	0.8169	0.8245	0.8527	0.8384
Approx-Skip-Chain	***0.8364***	0.8062	***0.8210***	***0.8614***	0.8651	***0.8632***

Jagannatha and Yu. 2016. Structured prediction models for RNN based sequence labeling in clinical text. EMNLP 2016.

Assertions

Table 1: Presence and Period Assertions.

ADE	Period	Presence
He has fever (caused by the drug)	Current	Present
He had fever (due to the drug)	History	Present
He has no fever (from the drug)	Current	Absent
His fever (caused by the drug is) resolved	History or current	Present or Absent
He has a fever, (possibly caused by the drug)	Current	Possible
He might have a fever	Current	Possible
If he is infected/(takes the drug), he will run a fever	Future	Conditional
He may develop a fever (with this drug)	Future	Hypothetical

Residual network for multi-task learning

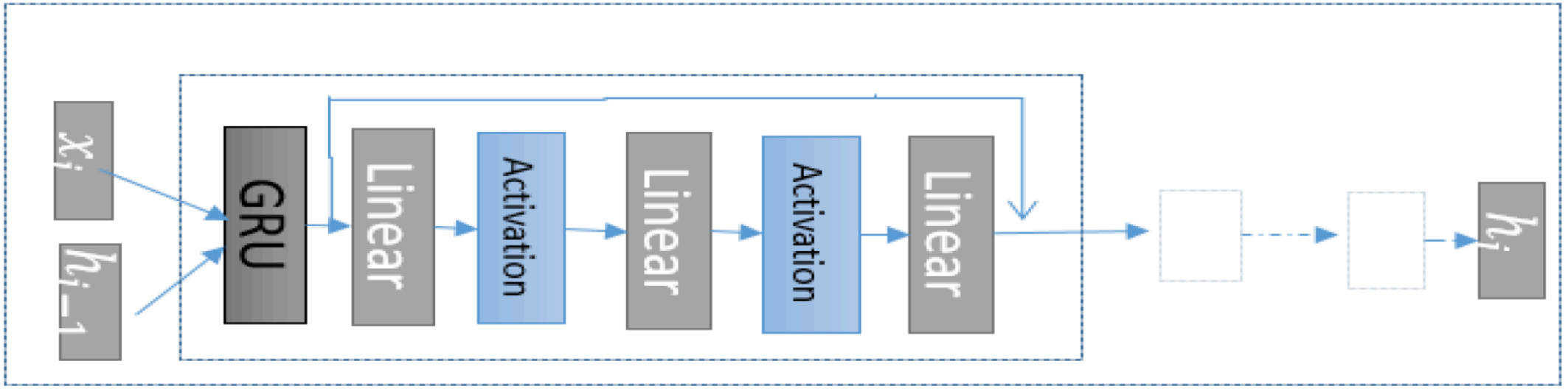


Figure 4: How the residual network functions in the proposed model

Table 4: Presence Assertion Results Comparison with Baseline. Each cell shows Recall/Precision/F-score

	SVM %	LSTM %	HNN Separate %	HNN joint%	HNN joint with extra feature%
Present	96.81/75.61/84.90	93.01/86.87/89.84	92.49/88.51/90.46	92.18/88.95/90.53	92.93/88.54/**90.69**
Absent	78.68/94.69/85.94	92.23/90.41/91.31	92.34/92.07/92.20	93.17/91.49/**92.33**	92.67/91.22/91.94
Not Patient	0/0/0	59.37/61.13/60.24	81.50/88.37/84.80	83.91/85.52/84.71	82.84/87.04/**84.89**
Conditional	0/0/0	0/0/0	2.78/2.08/2.38	2.78/2.13/2.41	2.78/2.56/**2.67**
Possible	0/0/0	2.31/2.22/2.26	27.94/28.48/28.21	31.11/36.84/**33.73**	38.05/24.76/30.00
Hypothetical	0/0/0	0/0/0/0	0/0/0	0/0/0	0/0/0
Micro Avg.	82.36/82.36/82.36	86.56/86.56/86.56	88.25/88.25/88.25	88.56/88.56/88.56	88.57/88.57/**88.57**
Macro Avg.	29.64/28.38/28.47	40.11/41.15/40.61	49.92/49.51/49.67	50.53/50.82/**50.62**	49.33/51.24/50.03

Table 5: Period Assertion Results Comparison with Baseline. Each cell shows Recall/Precision/F-score

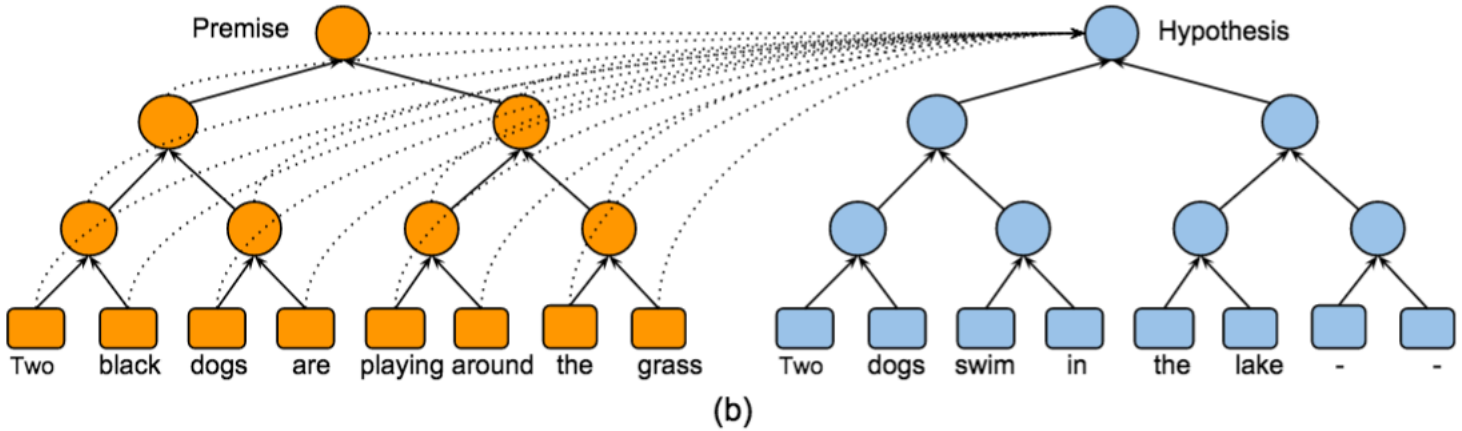
	SVM %	LSTM %	HNN Separate %	HNN joint%	HNN joint with extra feature%
History	0/0/0	31.76/33.42/32.57	66.08/66.45/66.27	67.72/68.93/**68.32**	63.92/68.55/66.16
Current	100/78.25/87.80	95.32/82.10/88.22	91.69/90.62/91.15	92.20/90.89/91.54	92.97/90.22/**91.57**
Future	0/0/0	5.69/7.01/6.28	33.60/49.60/**40.06**	31.17/43.89/36.45	31.17/53.24/39.32
Unknown	0/0/0	4.65/4.27/4.45	25.58/21.15/23.16	30.23/30.23/**30.23**	18.60/17.78/18.18
Micro Avg.	78.25/78.25/78.25	80.69/80.69/80.69	85.10/85.10/85.10	85.75/85.75/**85.75**	85.60/85.60/85.60
Macro Avg.	25.00/19.56/21.95	34.36/31.70/32.88	54.24/56.96/55.16	55.33/58.49/**56.64**	51.67/57.45/53.81

Four New Deep Learning Models

- Neural Tree Indexers (EACL, 2017)
- Neural Semantic Encoders (EACL, 2017)
- Reasoning NN (ICLR, 2017)
- Meta Networks (ICML, 2017)

Neural Tree Indexers

- LSTM models learn from the sequence
- Syntactic tree structure (recursive) has shown improved performance
- However, syntactic tree structure may be difficult to obtain especially in EHR narratives and therefore we introduce Neural Tree Indexers



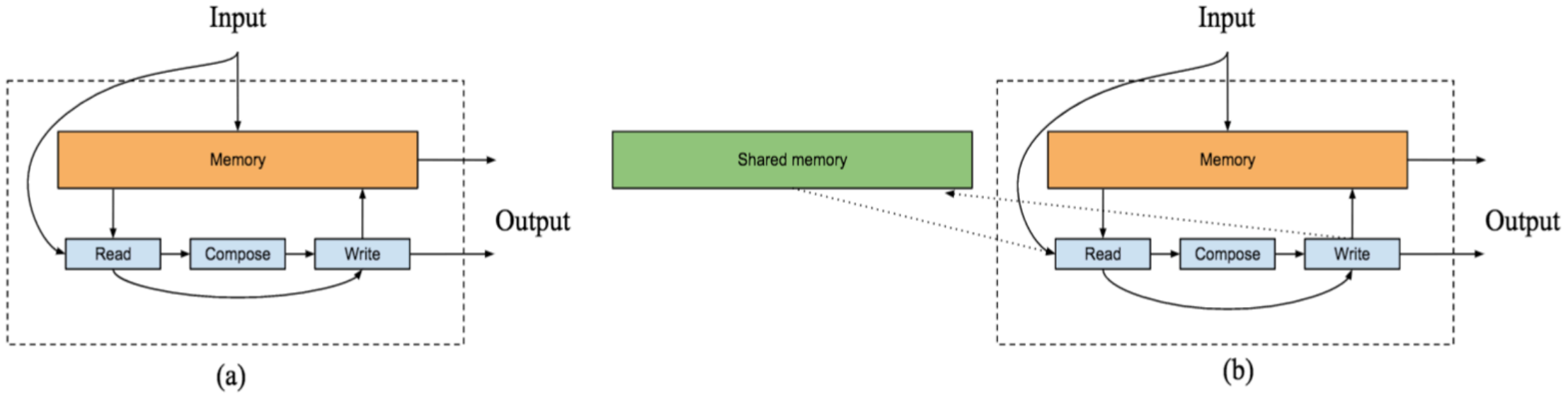
Munkhdalai and Yu. 2017. Neural Tree Indexers for Text Understanding. EACL 2017.

Model	d	Train	Test
Classifier with handcrafted features (Bowman et al., 2015a)	-	99.7	78.2
LSTMs encoders (Bowman et al., 2015a)	300	83.9	80.6
Dependency Tree CNN encoders (Mou et al., 2016)	300	83.3	82.1
NTI-SLSTM (Ours)	300	83.9	82.4
SPINN-NP encoders (Bowman et al., 2016)	300	89.2	83.2
NTI-SLSTM-LSTM (Ours)	300	82.5	83.4
LSTMs attention (Rocktäschel et al., 2016)	100	85.4	82.3
LSTMs word-by-word attention (Rocktäschel et al., 2016)	100	85.3	83.5
NTI-SLSTM node-by-node global attention (Ours)	300	85.0	84.2
NTI-SLSTM node-by-node tree attention (Ours)	300	86.0	84.3
NTI-SLSTM-LSTM node-by-node tree attention (Ours)	300	88.1	85.7
NTI-SLSTM-LSTM node-by-node global attention (Ours)	300	87.6	85.9
mLSTM word-by-word attention (Wang and Jiang, 2015)	300	92.0	86.1
LSTMN with deep attention fusion (Cheng et al., 2016)	450	88.5	86.3
Tree matching NTI-SLSTM-LSTM tree attention (Ours)	300	87.3	86.4
Decomposable Attention Model (Parikh et al., 2016)	200	90.5	86.8
Tree matching NTI-SLSTM-LSTM global attention (Ours)	300	87.6	87.1
Full tree matching NTI-SLSTM-LSTM global attention (Ours)	300	88.5	**87.3**

Table 1: Training and test accuracy on natural language inference task. d is the word embedding size.

Munkhdalai and Yu. 2017. Neural Tree Indexers for Text Understanding. EACL 2017.

Neural Semantic Encoders



Munkhdalai and Yu. 2017. Neural Semantic Encoders. EACL 2017.

Memory visualization

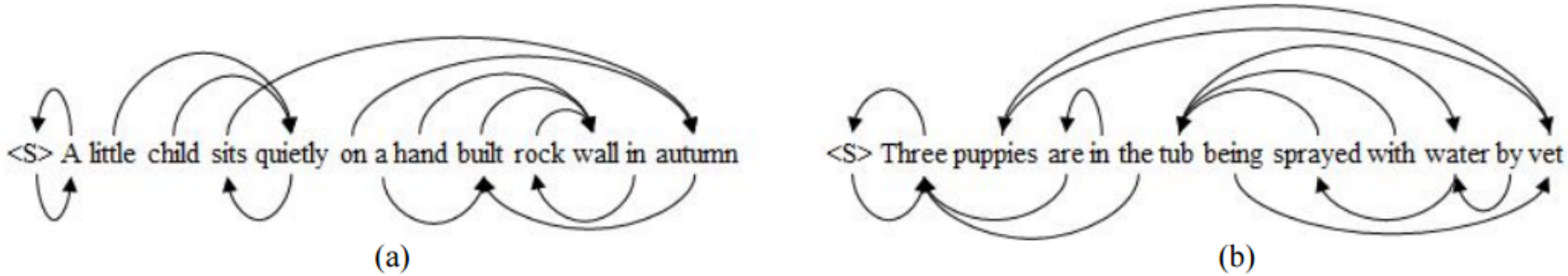


Figure 2: Word association or composition graphs produced by NSE memory access. The directed arcs connect the words that are composed via *compose* module. The source nodes are input words and the destination nodes (pointed by the arrows) correspond to the accessed memory slots. $< S >$ denotes the beginning of sequence.

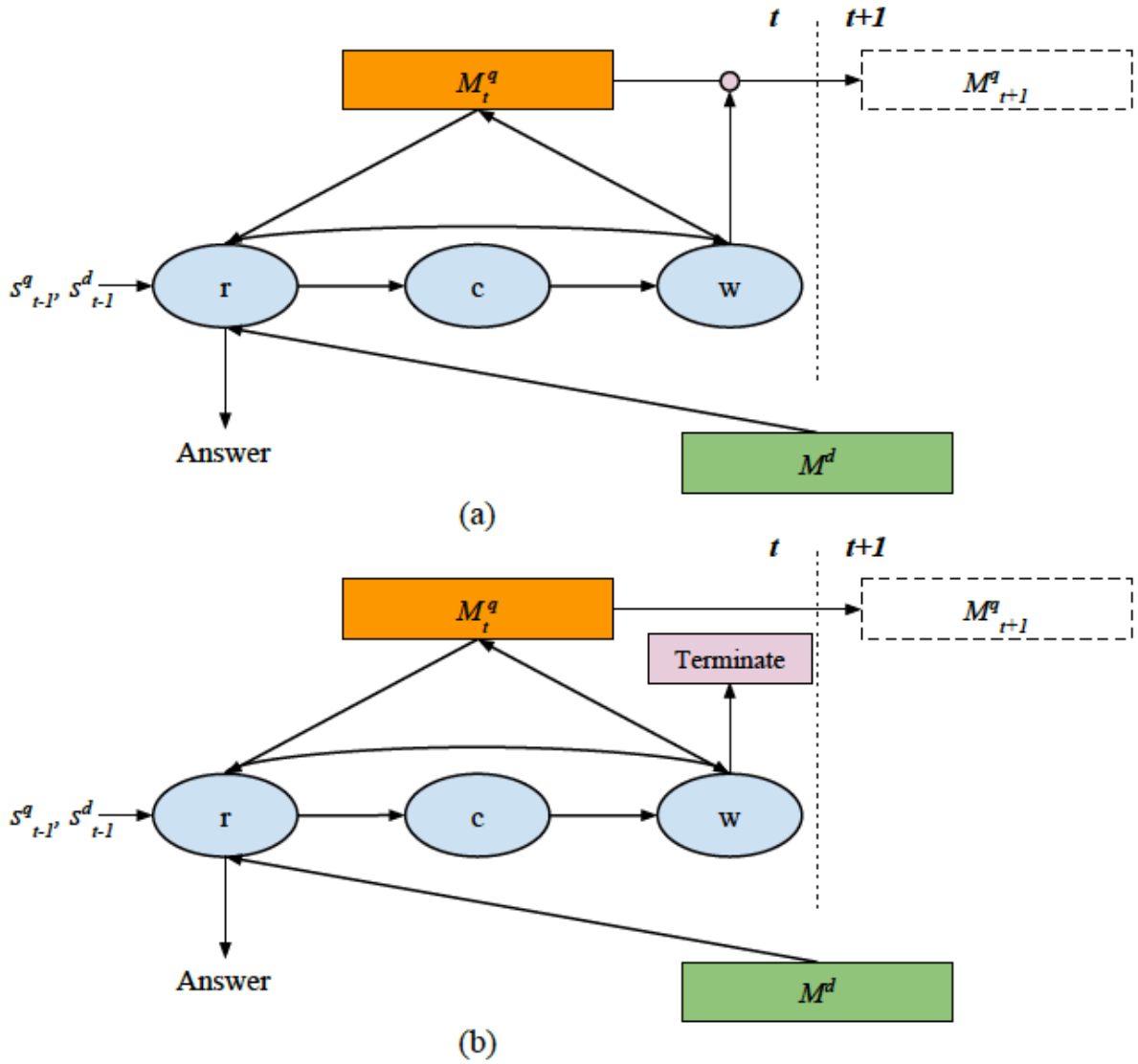
Results:
Answer sentence selection

- **Task:** select correct answer sentence from a candidate set to answer a question

Model	MAP	MRR
Classifier with features [22]	0.5993	0.6068
Paragraph Vector [23]	0.5110	0.5160
Bigram-CNN [24]	0.6190	0.6281
3-layer LSTM [25]	0.6552	0.6747
3-layer LSTM attention [25]	0.6639	0.6828
NASM [25]	0.6705	0.6914
MMA-NSE attention	**0.6811**	**0.6993**

Munkhdalai and Yu. 2017. Neural Semantic Encoders. EACL 2017.

Multi-step reasoning



Munkhdalai T and Yu H. Reasoning with Memory Augmented Neural Networks for Language Comprehension. ICLR 2017.

Results

Model	CBT-NE		CBT-CN	
	dev	test	dev	test
Human (context + query) (Hill et al., 2015)	-	81.6	-	81.6
LSTMs (context + query) (Hill et al., 2015)	51.2	41.8	62.6	56.0
MemNNs (window mem. + self-sup.) (Hill et al., 2015)	70.4	66.6	64.2	63.0
AS Reader (Kadlec et al., 2016)	73.8	68.6	68.8	63.4
GA Reader (Dhingra et al., 2016)	74.9	69.0	69.0	63.9
EpiReader (Trischler et al., 2016)	75.3	69.7	71.5	67.4
IAA Reader (Sordoni et al., 2016)	75.2	68.6	72.1	69.2
AoA Reader (Cui et al., 2016)	77.8	72.0	72.2	69.4
MemNN (window mem. + self-sup. + ensemble) (Hill et al., 2015)	70.4	66.6	64.2	63.0
AS Reader (ensemble) (Kadlec et al., 2016)	74.5	70.6	71.1	68.9
EpiReader (ensemble) (Trischler et al., 2016)	76.6	71.8	73.6	70.6
IAA Reader (ensemble) (Sordoni et al., 2016)	76.9	72.0	74.1	71.0
NSE ($T = 1$)	76.2	71.1	72.8	69.7
NSE Query Gating ($T = 2$)	76.6	71.5	72.3	70.7
NSE Query Gating ($T = 6$)	77.0	71.4	73.0	**72.0**
NSE Query Gating ($T = 9$)	78.0	72.6	73.5	71.2
NSE Query Gating ($T = 12$)	77.7	72.2	**74.3**	71.9
NSE Adaptive Computation ($T = 2$)	77.1	72.1	72.8	71.2
NSE Adaptive Computation ($T = 12$)	**78.2**	**73.2**	74.2	71.4

Munkhdalai T and Yu H. Reasoning with Memory Augmented Neural Networks for Language Comprehension. ICLR 2017.

Meta Networks

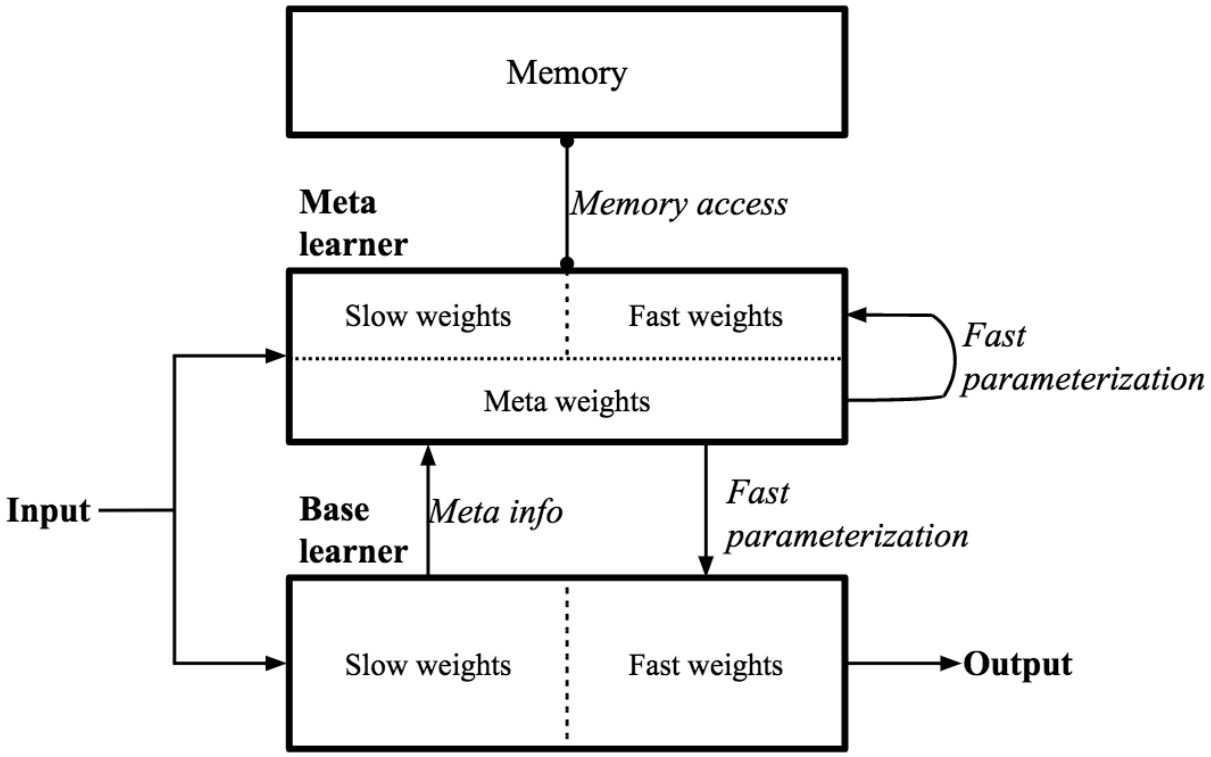


Figure 1. Overall architecture of Meta Networks.

Munkhdalai T and Yu H. Meta Networks. ICML 2017.

Meta Networks on One Shot Learning

Table 1. One-shot accuracy on Omniglot previous split

Model	**5-way**	**10-way**	**15-way**	**20-way**
Pixel kNN (Kaiser et al., 2017)	41.7	-	-	26.7
Siamese Net (Koch, 2015)	97.3	-	-	88.1
MANN (Santoro et al., 2016)	82.8	-	-	-
Matching Nets (Vinyals et al., 2016)	98.1	-	-	93.8
Siamese Net with Memory (Kaiser et al., 2017)	98.4	-	-	95.0
MetaNet-	98.4	98.32	96.68	96.13
MetaNet	**98.95**	**98.67**	**97.11**	**97.0**
MetaNet+	98.45	97.05	96.48	95.08

How Intelligent is a NLP System?

- Introducing Item Response Theory
  - Recall/Precision/Accuracy assume all items are equally difficult/easy
  - In reality, some items are easy and some items are hard
  - Use IRT as an alternative evaluation metrics

Lalor, Wu and Yu. 2016. Building an Evaluation Scale using Item Response Theory. EMNLP 2016.

Item Response Theory

$$p_{ij}(\theta_j) = c_i + \frac{1 - c_i}{1 + e^{-a_i(\theta_j - b_i)}}$$

j: Individual

i: Item

θ_j j's ability

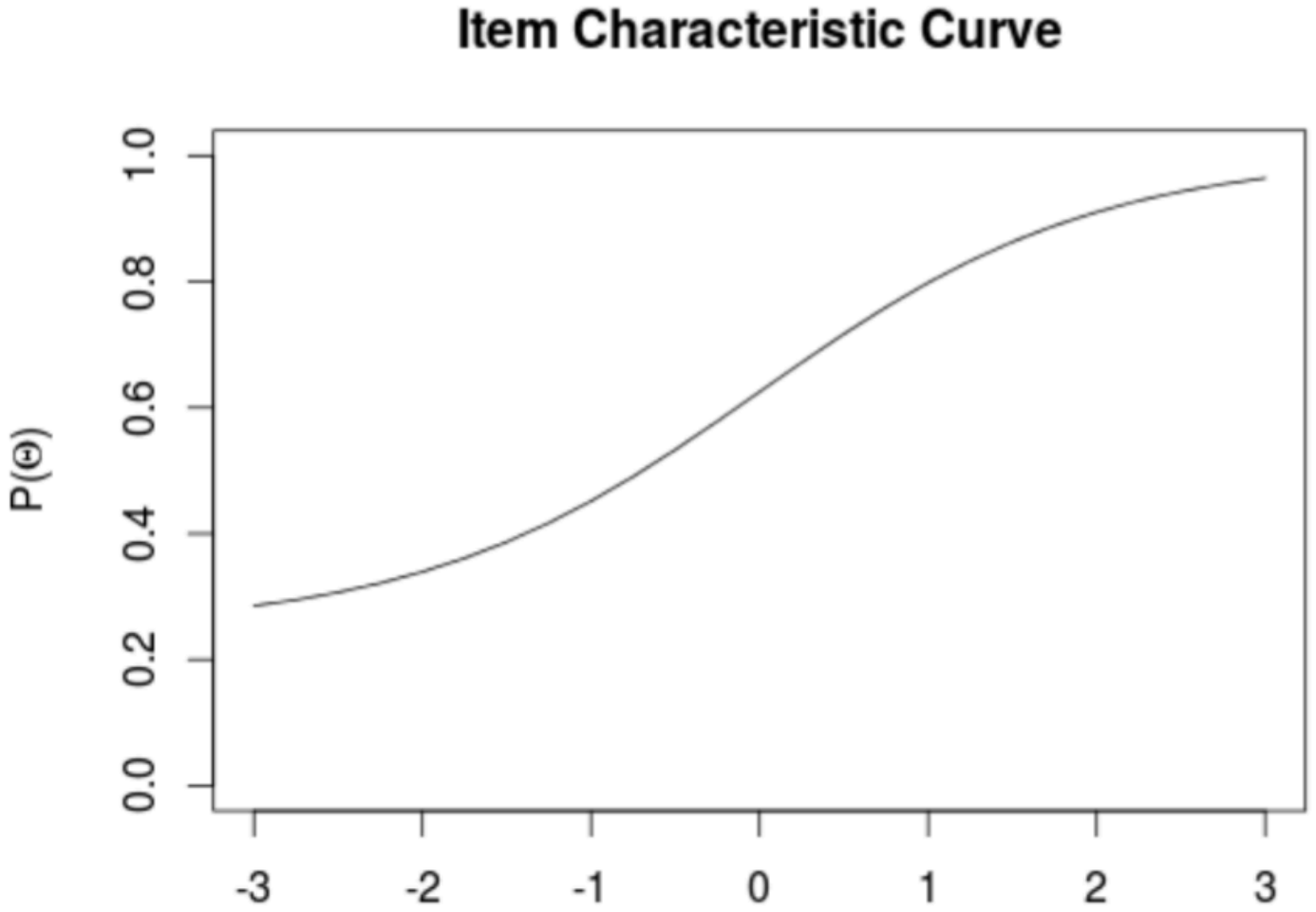
a_i: discrimination parameter

bi: difficulty

ci: guessing parameter

Lalor, Wu and Yu. 2016. Building an Evaluation Scale using Item Response Theory. EMNLP 2016.

Evaluation by Population Intelligence

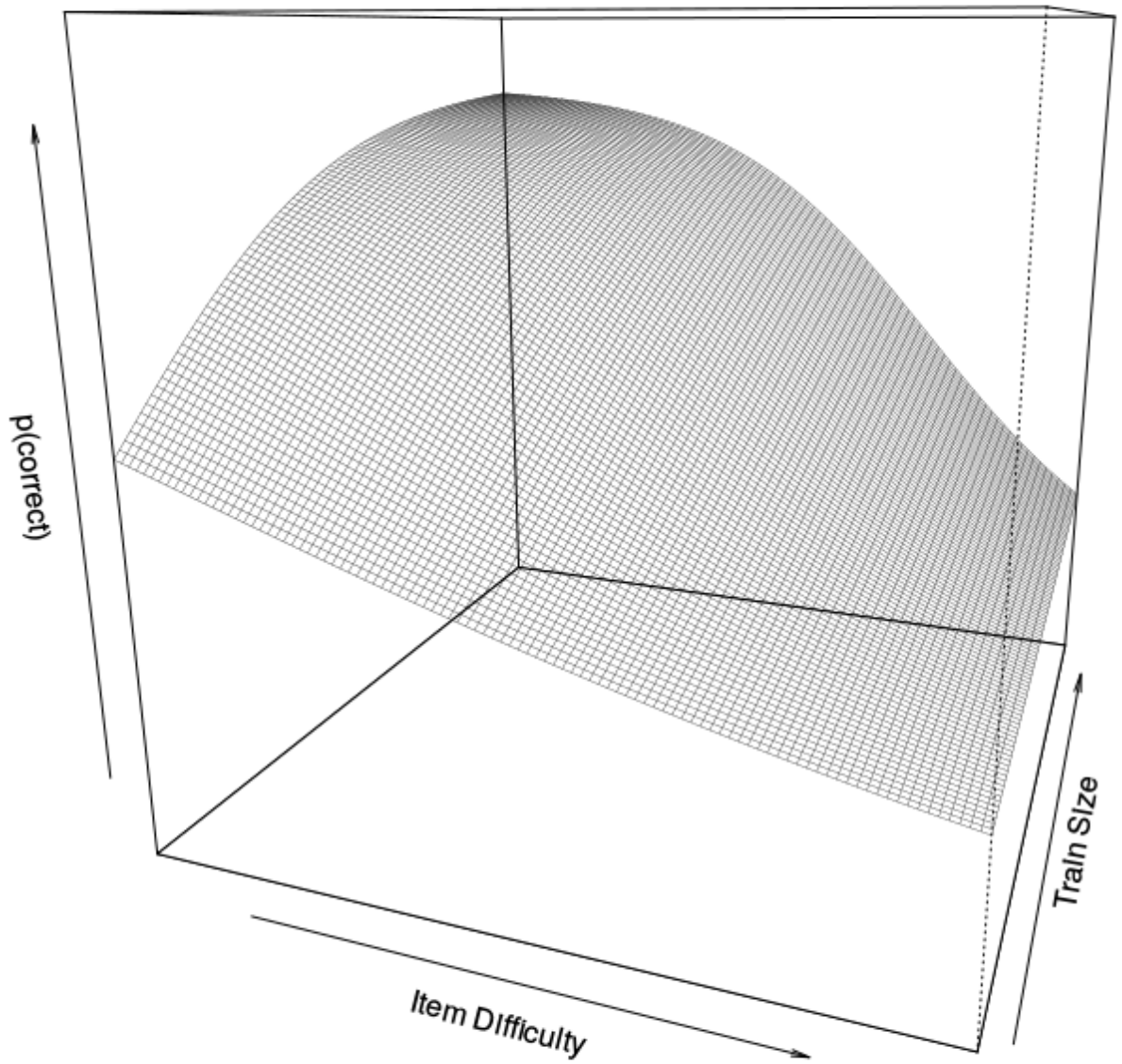


Lalor, Wu and Yu. 2016. Building an Evaluation Scale using Item Response Theory. EMNLP 2016.

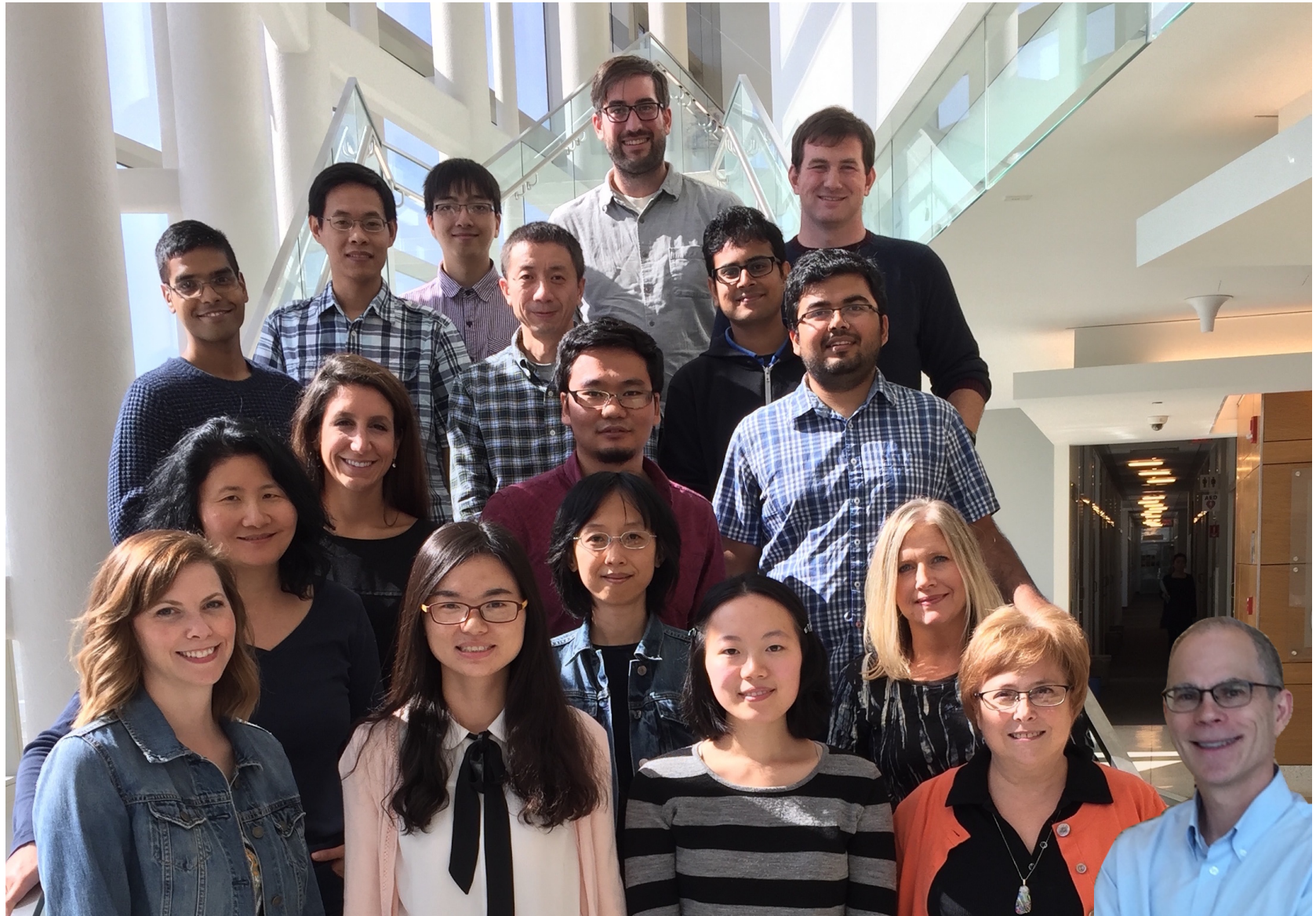
Separating Difficult Items from Easy Ones

Lalor, Wu and Yu. 2016. Building an Evaluation Scale using Item Response Theory. EMNLP 2016.

Learning with Easy and Difficult Items



Acknowledgment



Acknowledgment

National Institutes of Health
The Nation's Medical Research Agency

NIH National Heart, Lung, and Blood Institute

U.S. Department of Veterans Affairs

National Cancer Institute