

# Paraphrase Detection Ensemble – SemEval 2016 winner

1st PLACE IN THE ENGLISH SEMANTIC TEXTUAL SIMILARITY TASK

Katarzyna Pakulska, Barbara Rychalska, Krystyna Chodorowska,  
Wojciech Walczak and Piotr Andruszkiewicz

Samsung R&D Institute Poland, 2016

**SAMSUNG**

# Agenda

- What is SemEval?
- Paraphrase detection – main idea
- System base solution
  - Recursive auto encoders
  - Similarity matrix
  - Minimum polling
  - WordNet based module
- Ensemble Solution
  - Bidirectional GRU
  - Aligner
- Results
- Data sources



# What is SemEval?

- SemEval (Semantic Evaluation) is an ongoing series of evaluations of computational semantic analysis systems.
- Umbrella organization: SIGLEX, a Special Interest Group on the Lexicon of the Association for Computational Linguistics.

## Competition's tasks

### Track I. Textual Similarity and Question

#### Answering Track

Task 1: Semantic Textual Similarity:

A Unified Framework for Semantic Processing and Evaluation

Task 2: Interpretable Semantic Textual Similarity

Task 3: Community Question Answering

### Track II. Sentiment Analysis Track

...

### Track III. Semantic Parsing Track

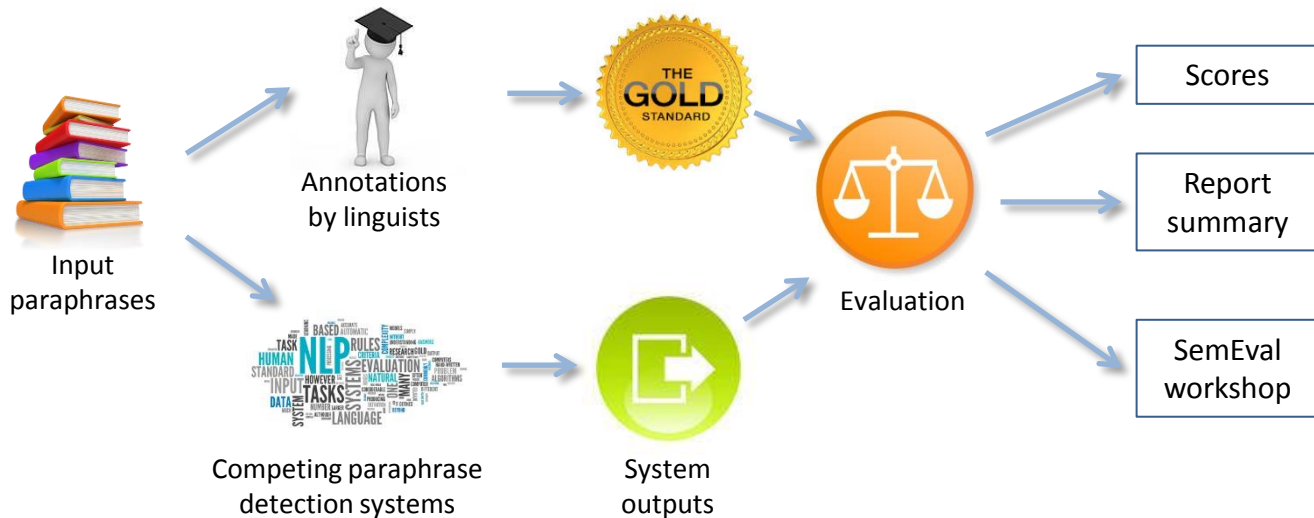
...

### Track IV. Semantic Analysis Track

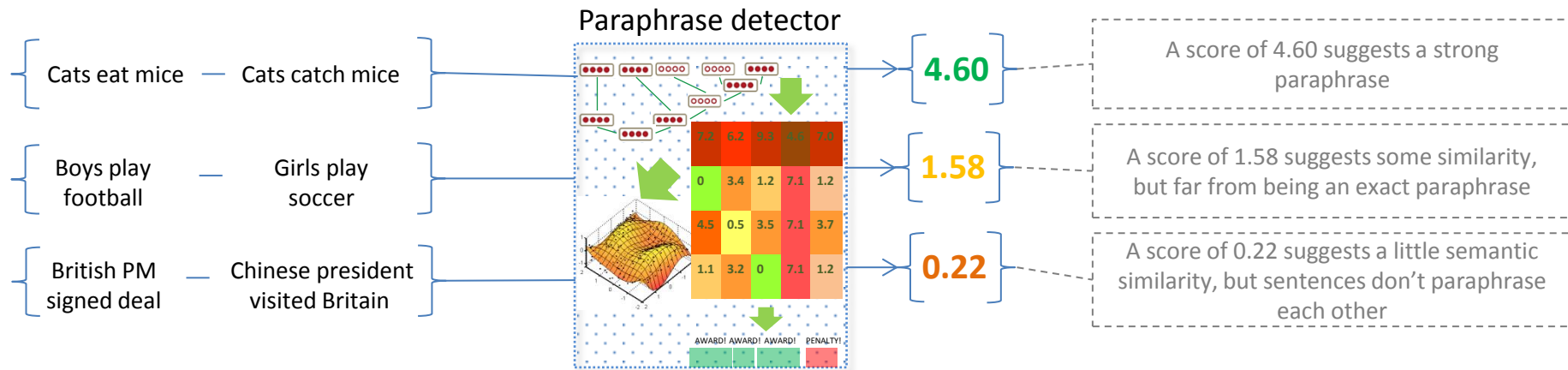
...

### Track V. Semantic Traxonomy Track

...

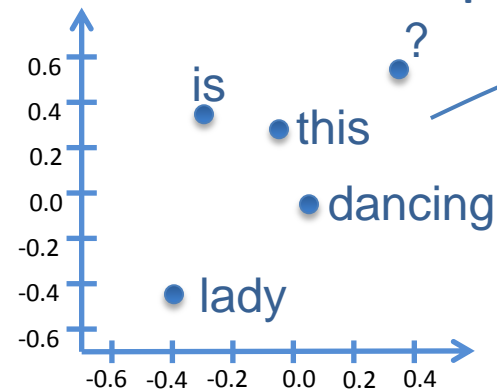


# Paraphrase detection



# Main idea: semantic similarity through word embeddings

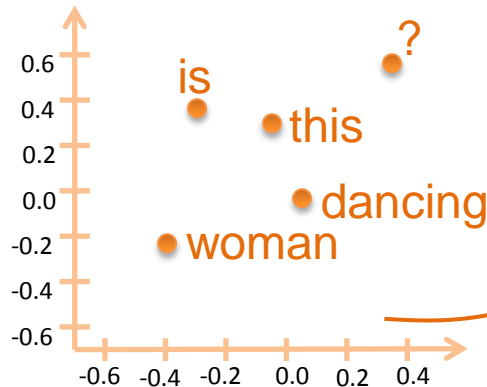
Word vector representation



is	[-0.3,	0.4]
this	[-0.1,	0.3]
lady	[-0.4,	-0.4]
dancing	[0.0,	[0.0]
?	[0.4,	0.6]

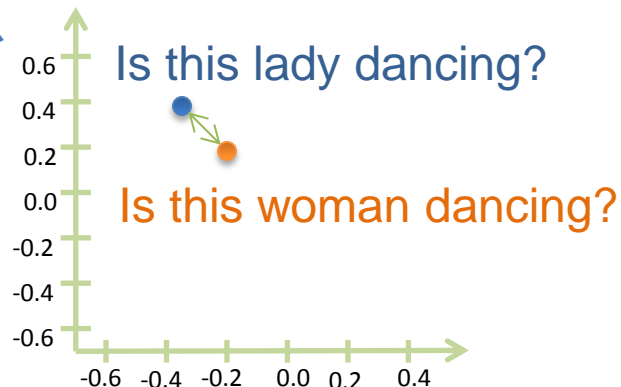
{ Is this lady dancing? }

Sentence vector representation



is	[-0.3,	0.4]
this	[-0.1,	0.3]
woman	[-0.4,	-0.2]
dancing	[0.0,	[0.0]
?	[0.4,	0.6]

{ Is this woman dancing? }

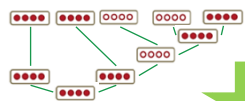


**Semantic similarity score: 5**  
(best level paraphrases)

# Paraphrase Detector: basic solution

Are two sentences similar?

Cats eat mice and fish



The cats catch mice

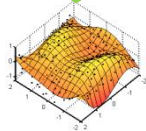


Cats eat mice and fish

The	2.2	6.2	9.3	4.6	7.0
cats	0	3.4	1.2	7.1	1.2
catch	4.5	0.5	3.5	7.1	3.7
mice	1.1	3.2	0	7.1	1.2

AWARD! AWARD! AWARD! PENALTY! AWARD! AWARD! AWARD!

Cats eat mice and fish The cats catch mice



3.45

STS competition aims to create a unified framework for measuring semantic similarity. A working evaluation tool must be able to detect that these two sentences have the same meaning.

The Recursive Auto Encoder takes unlabelled parse trees and word vectors as input and learns phrase features for each node. In the decoding part, the tree structure used to encode the sentence is mirrored.

A sentence similarity matrix is computed to generate similarity scores for two candidate sentences, using Euclidean distance as a measure of word-to-word similarity. The similarity scores are also counted for subtrees (not shown on the slide).

The WordNet-based module makes adjustments to the Euclidean distances between words represented as vectors based on:

- awarding pairs of words with positive semantic similarity;
- penalizing out-of-context words and disjoint similar concepts.

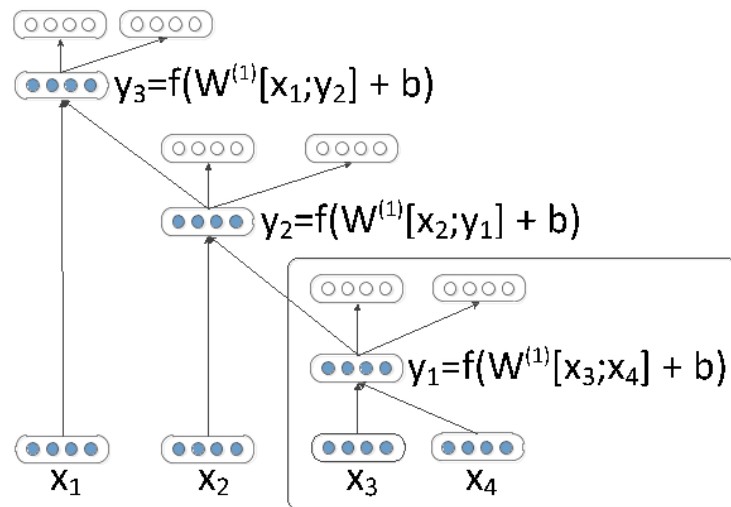
The WordNet-adjusted similarity matrices are converted to a matrix suitable for the Linear Support Vector Regression. The SVR model generates the final result.

The STS competition aimed at evaluating the sentences on the scale of 0 to 5, where 5 means perfect paraphrase. The score of 3.45 means that the sentences have a lot in common, but aren't an exact match.

# Recursive Auto Encoders

A recursive autoencoder reflects deep **grammatical tree structure** of a sentence and serves as a representation of **individual words** (other paraphrase detection methods use a bag-of-words approach – which causes loss of grammatical information).

Model	Acc.	F1
All Paraphrase Baseline	66.5	79.9
Rus et al. (2008) [16]	70.6	80.5
Mihalcea et al. (2006) [17]	70.3	81.3
Islam and Inkpen (2007) [18]	72.6	81.3
Qiu et al. (2006) [19]	72.0	81.6
Fernando and Stevenson (2008) [20]	74.1	82.4
Wan et al. (2006) [21]	75.6	83.0
Das and Smith (2009) [15]	73.9	82.3
Das and Smith (2009) + 18 Features	76.1	82.7
Unfolding RAE + Dynamic Pooling	<b>76.8</b>	<b>83.6</b>



*Illustration of an application of a recursive auto encoder to a binary tree. The nodes which are not filled are only used to compute reconstruction errors. A standard auto encoder (in box) is re-used at each node of the tree.*

Based on [2]

**„Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection”**

Richard Socher and Eric H. Huang and Jeffrey Pennington and Andrew Y. Ng and Christopher D. Manning

<http://www.socher.org/index.php/Main/DynamicPoolingAndUnfoldingRecursiveAutoencodersForParaphraseDetection>

# RAE: Natural Stanford parse tree

This approach was used in 2014, although not for paraphrase recognition. It was used for describing images with sentences.

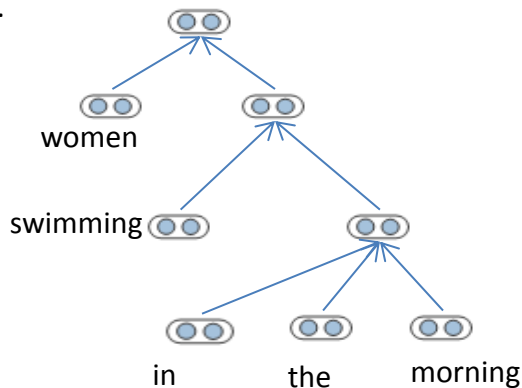
## Benefit:

Creating a deep net that can use authentic parse trees, which are inherently non-binary.

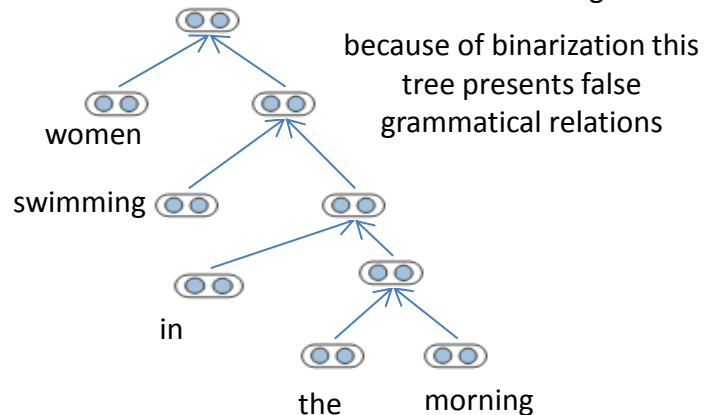
In basic approach, all trees had to be artificially binarized so that the network could be trained. Other tree structures damaged matrix dimensionalities during training.

Sentence:

**Women swimming in  
the morning**



**natural Stanford parse tree**



**artificially binarized Stanford  
parse tree**

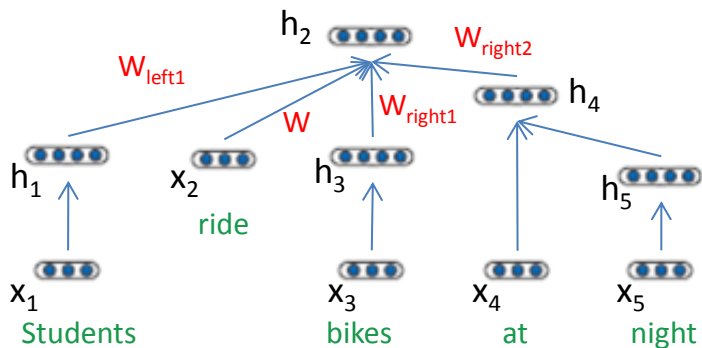
## How to make it work?

Multiply and add vectors during training instead of concatenating them. The modified network can use any tree structure.

See training procedure described in [3]: „Grounded Compositional Semantics For Finding And Describing Images With Sentences” Richard Socher, Andrej Karpathy, Quoc V. Le, Christopher D. Manning, Andrew Y. Ng.

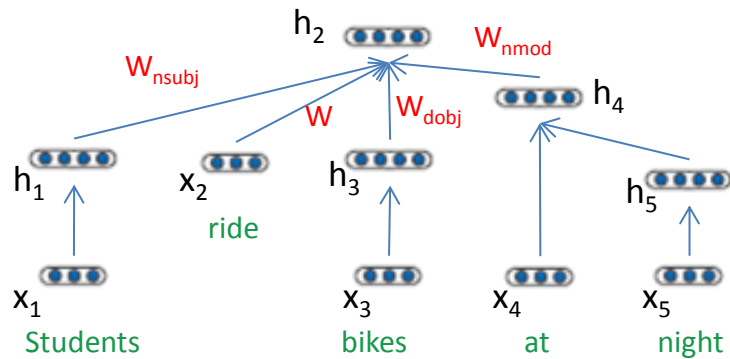


# RAE: SDTRNN



$$h_2 = f(W_{\text{left1}} * h_1 + W_{\text{right1}} * h_3 + W_{\text{right2}} * h_4 + W * x_2)$$

**Dependency Tree Recursive Neural Networks (DTRNN):** weight matrices created for ordering nodes

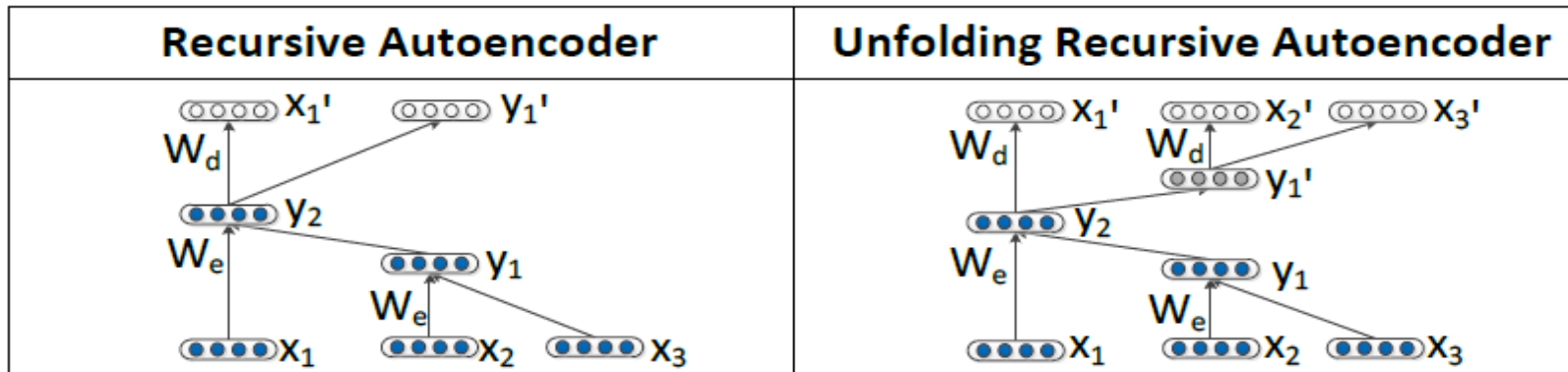


$$h_2 = f(W_{\text{nsubj}} * h_1 + W_{\text{dobj}} * h_3 + W_{\text{nmod}} * h_4 + W * x_2)$$

**Semantic Dependency Tree Recursive Neural Networks (SDTRNN):** weight matrices created for dependency relations

<i>Sentences Similarity for Image</i>		<i>Image Search</i>		<i>Describing Images</i>	
Model	Mean Rank	Model	Mean Rank	Model	Mean Rank
Random	101.1	Random	52.1	Random	92.1
BoW	11.8	BoW	14.6	BoW	21.1
CT-RNN	15.8	CT-RNN	16.1	CT-RNN	23.9
Recurrent NN	18.5	Recurrent NN	19.2	Recurrent NN	27.1
kCCA	10.7	kCCA	15.9	kCCA	18.0
DT-RNN	11.1	DT-RNN	13.6	DT-RNN	19.2
SDT-RNN	<b>10.5</b>	SDT-RNN	<b>12.5</b>	SDT-RNN	<b>16.9</b>

# RAE: Unfolding recursive auto encoder

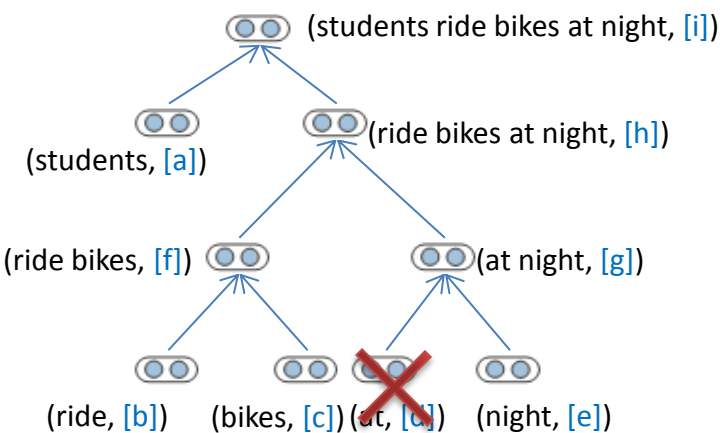
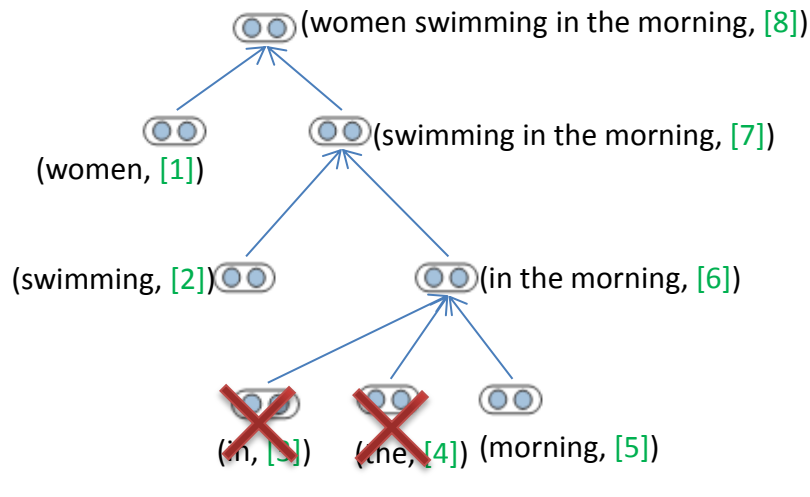


$$E_{rec}(y_{(i,j)}) = ||[x_i; \dots; x_j] - [x'_i; \dots; x'_j]||^2$$

- decoding at each node
- decoding of **only 2 intermediate children**

- decoding at each node
- decoding of a **full subtree**
- does not propagate errors of intermediate nodes

# Similarity matrix



1. Ordering the nodes in pooling matrices indicates level order with sorting using node depth information for prioritization nodes in a pooling module.

2. A list of stop words was created for leaves, which eliminated some words from the pooling module (for example, words such as „a”, „the” and some punctuation characters). Phrases in parse trees for the deep net were unchanged.

leaves

ED[1,a]	ED[1,b]	ED[1,c]	ED[1,e]	ED[1,f]	ED[1,g]	ED[1,h]	ED[1,i]
ED[2,a]	ED[2,b]	ED[2,c]	ED[2,e]	ED[2,f]	ED[2,g]	ED[2,h]	ED[2,i]
ED[5,a]	ED[5,b]	ED[5,c]	ED[5,e]	ED[5,f]	ED[5,g]	ED[5,h]	ED[5,i]
ED[6,a]	ED[6,b]	ED[6,c]	ED[6,e]	ED[6,f]	ED[6,g]	ED[6,h]	ED[6,i]
ED[7,a]	ED[7,b]	ED[7,c]	ED[7,e]	ED[7,f]	ED[7,g]	ED[7,h]	ED[7,i]
ED[8,a]	ED[8,b]	ED[8,c]	ED[8,e]	ED[8,f]	ED[8,g]	ED[8,h]	ED[8,i]

leaves

↓

sentence to sentence

Where ED -> Euclidean Distance

# Minimum pooling

0.54	2.19	5.26	8.25	6.29	9.26	4.73	5.32
8.28	3.72	1.62	6.32	7.54	5.27	7.75	7.95
9.07	9.43	6.43	3.14	5.13	7.86	9.36	6.73
6.53	6.26	7.24	4.26	1.26	4.82	4.89	9.47
7.27	5.84	5.26	8.64	5.37	2.84	8.46	3.65
6.39	9.29	8.53	7.85	8.15	5.93	3.94	3.78

Assume that the pooling window is 4, then system creates grid.

2.19	5.26	6.29	4.73
3.72	1.62	5.27	7.75
6.26	3.14	1.26	4.89
5.84	5.26	2.84	3.65

Replace each  $c$  cell of the grid with minimum values from  $c$ .

**Additional Features:** three features represent number similarity, absolute difference in length, percentage of similarity, adjustment roots, score from aligner's classifier, cosine measure between roots, negation comparison, three WordNet features represent the respective penalties and awards.

# Differences in results relative to pooling window

RAE Model: SDTRNN,  
Train examples: 10000 ,  
Iteration: 50 ,  
Words vectors size: 50.

Pooling window size	MSRpar	MSRvid	SMTeuroparl	OnWN	SMTnews
3	0.596	0.401	0.245	0.338	0.33
5	<b>0.603</b>	0.731	0.445	0.467	0.335
7	0.554	0.776	0.428	<b>0.528</b>	<b>0.413</b>
9	0.56	0.773	0.447	0.498	<b>0.413</b>
11	0.503	<b>0.781</b>	0.459	0.463	0.410
13	0.488	0.779	<b>0.466</b>	0.443	0.408
15	0.457	0.772	0.433	0.436	0.322

# WordNet based module

**Problem:** Recursive Autoencoders focus on distributional similarity, without accounting for semantic similarities.

**Solution:** Use WordNet to adjust RAE scores with awards and penalties based on the semantic similarity of pairs of words [4].

## **Approaches:**

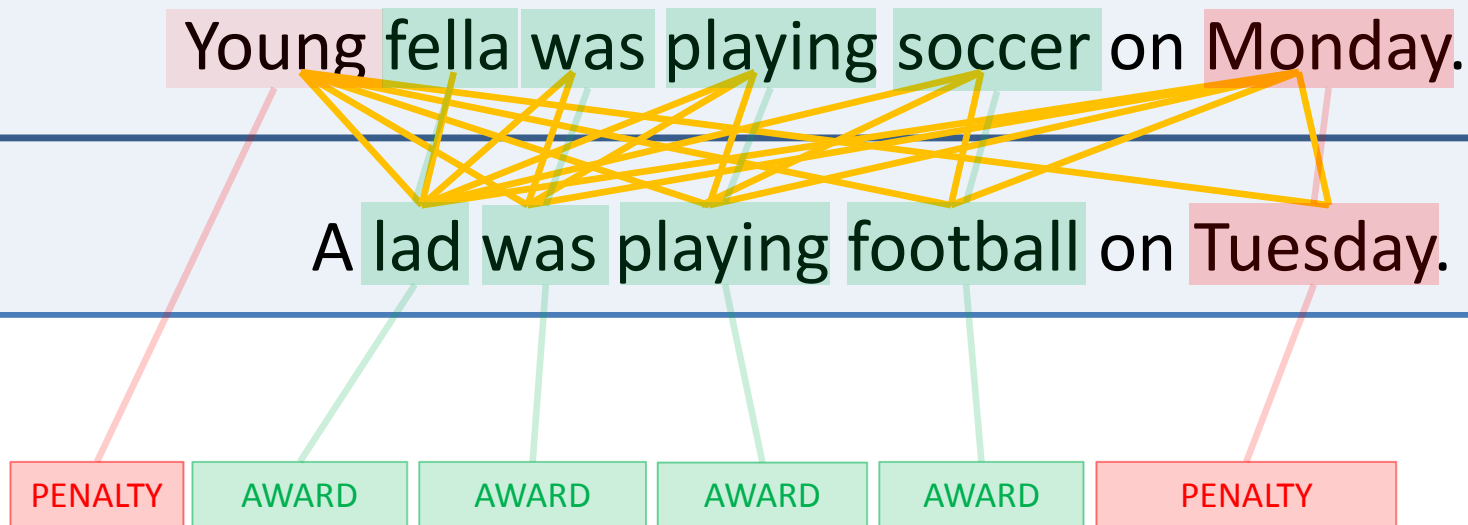
- Awarding pairs of words with positive semantic similarity
- Penalizing out-of-context words, antonyms and disjoint concepts
- Propagating scores on higher nodes of dependency trees

# Awarding pairs of words with positive semantic similarity

1. Extracting semantic relations for nouns, verbs, adjectives and adverbs.
2. Using separate similarity measures for nouns, verbs, adjectives and adverbs , based on the following features (in terms of increased similarity, descending):
  - The two items being synonyms;
  - Sharing a significant sense (one word refers to the main meanings of the other word);
  - Being similar (adjectives and adverbs; eg.: "abridged" is similar to "shortened");
  - Being hypernyms or two-link hypernyms (nouns and verbs);
  - Sharing any common meaning;
  - Being derivationally related;
  - Being enclosed in the glosses of the other word's meanings.
3. Replacing measures such as Lin (information-content based) or Path (path-length based). Our measure is more conservative (analysis of Lin and Path suggest that these measures are returning positive scores for unrelated inputs).

# Penalizing

- Out-of-context words: words not paired in the two input sentences. Three strategies:
  - Penalize all recognizable parts of speech
  - Penalize only nouns
  - Penalize only physical objects
- Antonyms: words with an opposite meaning (i.e. good – bad)
- Disjoint concepts: words with a disjoint meaning (i.e. Monday – Tuesday). A disjoint set was based on the hypernym hierarchy. If two words have a common direct hypernym, they are treated as disjoint concepts (e.g. both Monday and Tuesday have „weekday” as a common hypernym).

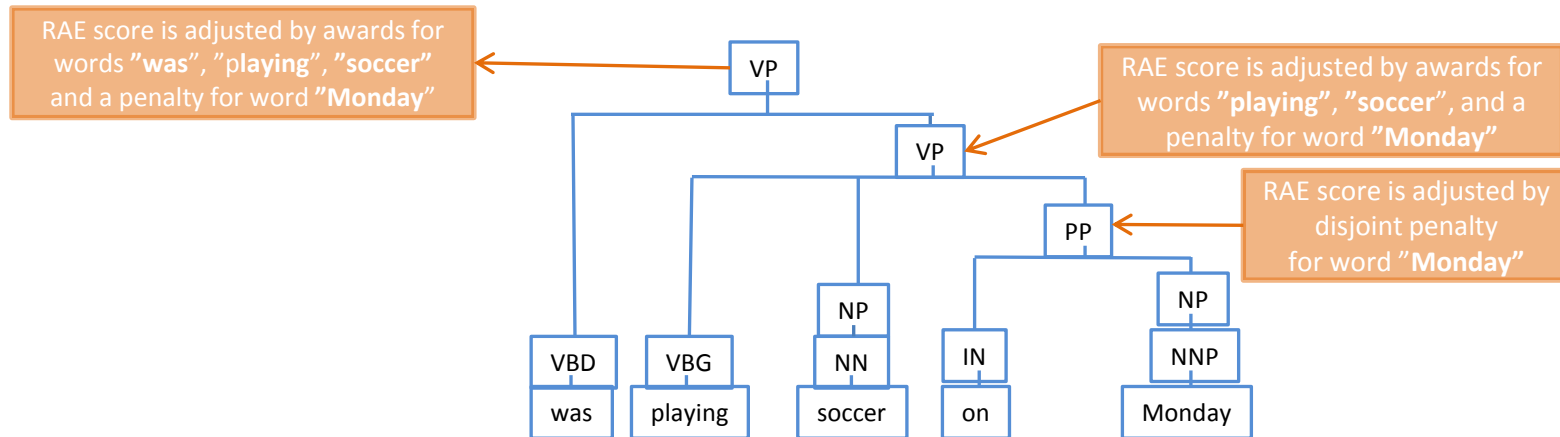




# Propagating scores on higher nodes of dependency trees

- RAE calculates its score for all subtrees in a sentence, while WordNet awards and penalties are calculated for words. Thus, to use WordNet's scores on all subtrees, the awards and penalties have to be propagated.
- Given a subtree, all awards and penalties for leaves in a subtree are first divided by the depth of the tree, and then added up.
- Awards and penalties are divided by their depth relative to the root of the subtree to account for their importance in the more complex subtrees.

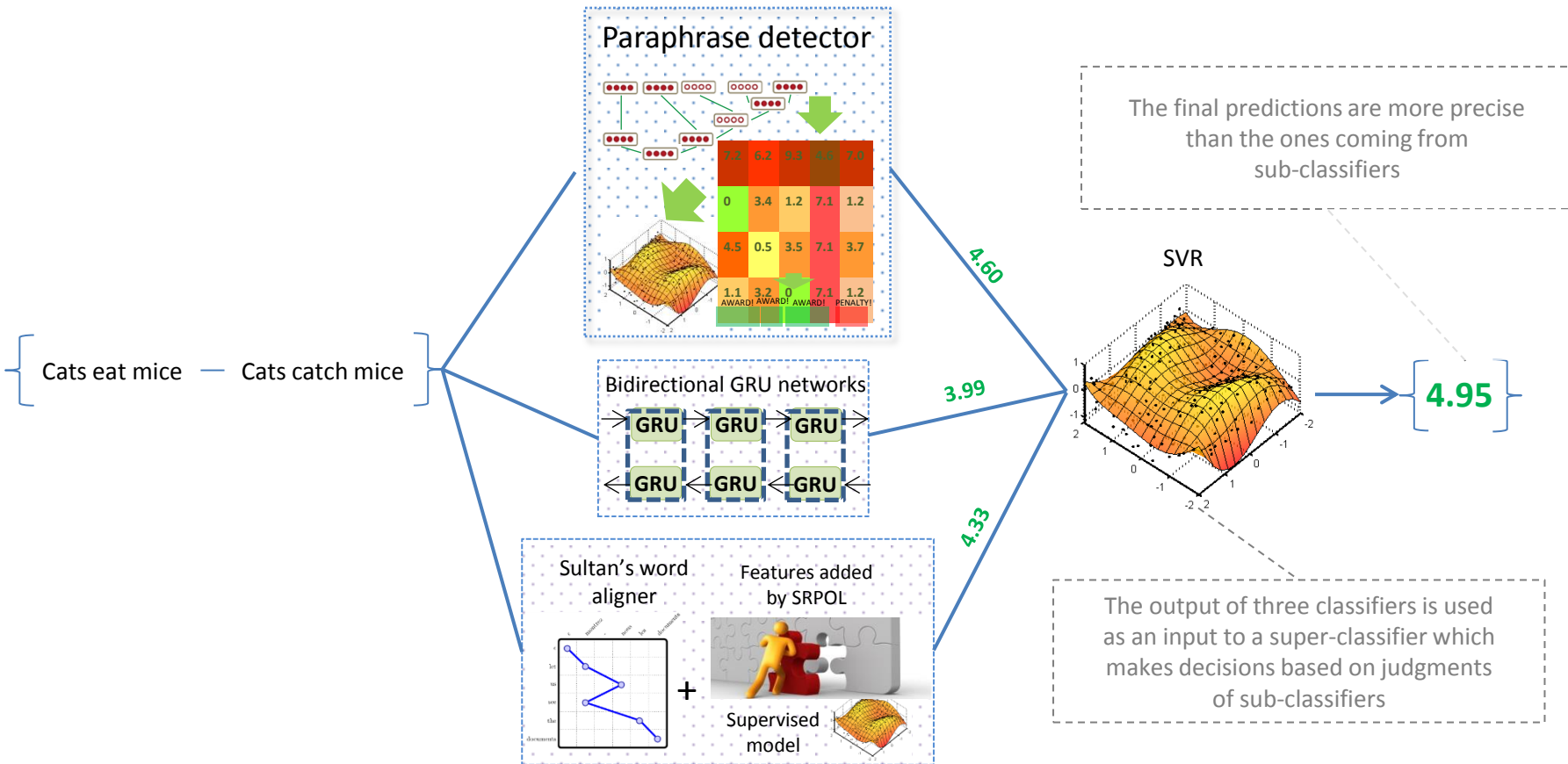
An example for a subtree related to the part of the sentence: „was playing soccer on Monday” is presented below:



# Results for different part of Paraphrase Detector

	Answers forums	Answers students	Belief	Headlines	Images	Average
RAE	0.4724	0.7066	0.6109	0.706	0.7469	0.6486
RAE + WordNet	0.4916	0.7185	0.6002	0.7115	0.7797	0.6603
RAE + WordNet + Propagation	0.5404	0.7085	0.6418	0.7114	0.7952	0.6794
Full Solution	0.6836	0.7679	0.7517	0.8315	0.8625	0.7794

# Paraphrase Detector: ensemble solution

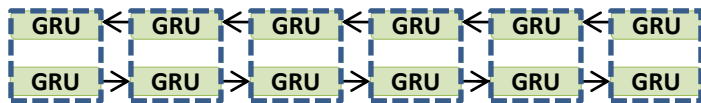


# Bidirectional GRU-based classifier

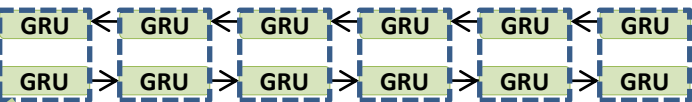
Lookup table

Bidirectional GRU networks

Output module



A bird lands in the water



A boat floats in the water



Hidden layer

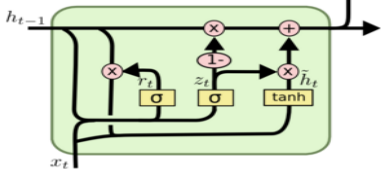


Softmax over possible scores



0.6

## Gated Recurrent Unit



$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t])$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Hadamard products and differences of vectors from final states of the networks

Implementation in Lua/Torch, code adapted from Kai ShengTai et al.:

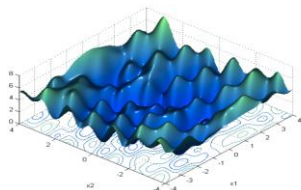
<https://github.com/stanfordnlp/treelstm>

# Bidirectional GRU: performance

Other architectures (LSTM, bidirectional LSTM, Binary Tree-LSTM, Child-Sum Tree-LSTM) were also tried, but bidirectional GRU yielded best results

## Hyperparameter configuration (found by random search)

- Number of deep layers: **1**
- Embeddings dimensionality: **300**
- GRU unit hidden state dimensionality: **150**
- Output module hidden dimensionality: **25**
- L2 regularization strength: **0.00005**
- Parameter learning rate: **0.1**
- Embedding learning rate: **0.0**
- Dropout on output module layers: **0.0**
- Dropout on input layer: **0.0**
- Minibatch size: **50**



## Performance (trained on 75% split)

Answer-answer	39,4%
Question-question	32,9%
Headlines	66,9%
Plagiarism	74,8%
Postediting	70,0%



Align identical  
word sequences

- 

## Align named entities

- 

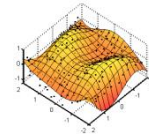
## Align content words using dependencies









## Align content words using surrounding words






## Scoring



warm   cold 

big   small 

fast   slow 

## Negation



# Results & who was beaten

## Companies:

- Toyota Technological Institute
- RICOH
- Mayo Clinic
- IHS Markit

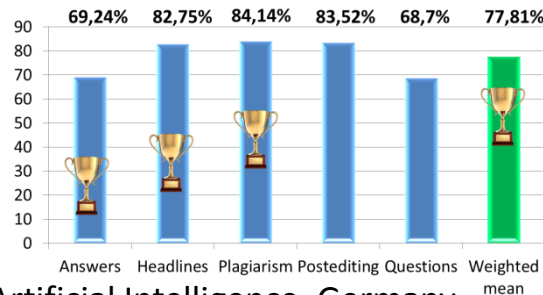
## Public research institutions:

- German Research Center for Artificial Intelligence, Germany
- National Centre for Text Mining, UK
- Institute of Software, Chinese Academy of Sciences, China

## Universities:

- University of Colorado Boulder, USA
- University of Texas, Arlington, USA
- University of Sheffield, UK
- University of Sussex, UK
- Universität des Saarlandes, Germany
- Heinrich Heine University Düsseldorf, Germany
- University of Madrid, Spain
- Dublin City University, Ireland
- Beijing Institute of Technology, China

...and others!



## Top 10 results during SemEval 2016

Place	Team	Overall mean
1	Samsung R&D Poland: ensemble 1	77.8%
2	University of West Bohemia, Czech Republic	75.7%
3	Mayo Clinic, USA	75.6%
4	Samsung R&D Poland: ensemble 2	75.4%
5	East China Normal University, China	75.1%
6	The National Centre for Text Mining, UK	74.8%
7	Univeristy of Maryland, USA Toyota Technological Institute, USA University of Waterloo, Canada	74.2%
8	University of Massachusetts Lowell, USA	73.8%
9	Mayo Clinic NLP Team	73.569%
10	Samsung R&D Poland: basic solution	73.566%

# Data Sources

## Training- and testsets

### Open American National Corpus:

- contains i.e. news articles and journals
- 130.000 sentences were sampled from this dataset and used in training of the deep net

### Microsoft Research Paraphrase Corpus:

- 5801 pairs of sentences (from web news)
- 2 human judges with binary judgment, disagreements resolved by 3rd judge
- 3900 (67%) pairs judged semantically equivalent
- most commonly used in reference papers
- already divided into trainset and testset – good for comparing results with other authors

### Gold Standard of SemEval 2012-2015 Task : Semantic Textual Similarity for English

- [http://ixa2.si.ehu.es/stswiki/index.php/Main\\_Page](http://ixa2.si.ehu.es/stswiki/index.php/Main_Page)
- the gold standard contains a score between 0 and 5 for each pair of sentences. The gold standard file consist of one single field per line: - a number between 0 and 5
- will be used in preparation for the contest

## Lexical resources

### WordNet for English language

POS	Unique Strings	Synsets	Total Word-Sense Pairs
Noun	117798	82115	146312
Verb	11529	13767	25047
Adjective	21479	18156	30002
Adverb	4481	3621	5580
Totals	155287	117659	206941



# Bibliography:

1. **"Samsung Poland NLP Team at SemEval-2016 Task 1: Necessity for diversity; combining recursive autoencoders, WordNet and ensemble methods to measure semantic similarity"**, Barbara Rychalska, Katarzyna Pakulska, Krystyna Chodorowska, Wojciech Walczak and Piotr Andruszkiewicz
2. **"Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection"** Richard Socher and Eric H. Huang and Jeffrey Pennington and Andrew Y. Ng and Christopher D. Manning
3. **"Grounded Compositional Semantics For Finding And Describing Images With Sentences"**, Richard Socher, Andrej Karpathy, Quoc V. Le, Christopher D. Manning, Andrew Y. Ng.
4. **"Semantic Textual Similarity Systems"**, Lushan Han, Abhay Kashyap, Tim Finin, James Mayfield and Jonathan Weese,
5. **"WordNet: Similarity - Measuring the Relatedness of Concepts"**, Ted Pedersen, Siddharth Patwardhan, Jason Michelizzi
6. **"WordNet: A Lexical Database for English"**, George A. Miller (1995). Communications of the ACM Vol. 38, No. 11: 39-41.

# Bibliography:

6. **"WordNet: An Electronic Lexical Database"**, Christiane Fellbaum (1998, ed.) . Cambridge, MA: MIT Press.
7. **"Samsung: Align-and-Differentiate Approach to Semantic Textual Similarity"**, Lushan Han, Justin Martineau, Doreen Cheng, Christopher Thomas
8. **"DLS@CU: Sentence Similarity from Word Alignment"** Arafat Sultan, Steven Bethard, Tamara Sumner
9. **"Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks"** Kai Sheng Tai, Richard Socher, Christopher D. Manning
10. **"ExB Themis: Extensive Feature Extraction from Word Alignments for Semantic Textual Similarity"** Christian Hanig, Robert Remus, Xose De La Puente

Thank you!

Questions?