Representation Learning for Discourse Parsing

Jacob Eisenstein and Yangfeng Ji

Georgia Institute of Technology

December 12, 2014

Discourse in natural language

What makes...

- an interesting story?
- a good joke?
- a persuasive argument?

Discourse in natural language

What makes...

- an interesting story?
- a good joke?
- a persuasive argument?

What do we focus on in NLP?

- Individual sentences (e.g., parsing, translation)
- Bag-of-words representations of documents (e.g., topic models, sentiment analysis)

Discourse in natural language

What makes...

- an interesting story?
- a good joke?
- a persuasive argument?

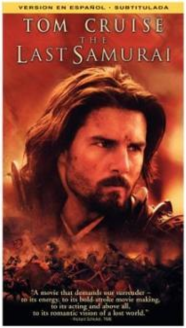
What do we focus on in NLP?

- Individual sentences (e.g., parsing, translation)
- Bag-of-words representations of documents (e.g., topic models, sentiment analysis)

Discourse is the missing link between micro-level and macro-level linguistic phenomena.

Example: discourse and sentiment

It could have been a **great** movie. It could have been **excellent**, and to all the people who have forgotten about the older, **greater** movies before it, will think that as well. It does have **beautiful** scenery, some of the **best** since Lord of the Rings. The acting is **well** done, and I really **liked** the son of the leader of the Samurai. He was a **likeable** chap, and I **hated** to see him die... But, other than all that, this movie is nothing more than hidden **rip-offs**.



Discourse and sentiment

Voll and Taboada [VT07]:

- Annotated discourse structure (RST) improves sentiment analysis...
- ... but automatically-parsed discourse structure makes it worse!

Why is discourse hard?

- Discourse relations are **semantic**:
  - Montreal is a bilingual city
  - (Because) they speak French and English
- Typical solution is bilexical features, e.g., $\langle$speak,bilingual$\rangle$
- Discourse-annotated datasets are way too small for this to work.

Why is discourse hard?

- Discourse relations are **semantic**:
  - Montreal is a bilingual city
  - (Because) they speak French and English
- Typical solution is bilexical features, e.g., $\langle$speak,bilingual$\rangle$
- Discourse-annotated datasets are way too small for this to work.
- **Representation learning can help**, by inducing distributed models of discourse semantics.

This talk

Representation learning for two discourse structures:

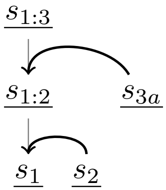
- **Rhetorical structure theory**: learn to parse discourse into trees, while jointly learning word representations [JE14b].
- **Penn Discourse Treebank**: learn compositional operators for distributed semantics [JE14a].

Yangfeng Ji

Rhetorical structure theory

In RST, discourse relations compose elementary units into a tree [MT88].

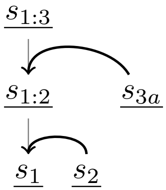
- *s*1: Montreal is a bilingual city.
- *s*2: They speak French and English.
- *s*3*a*: This makes it an interesting place to visit.

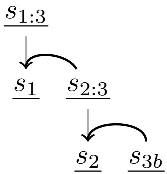


Rhetorical structure theory

In RST, discourse relations compose elementary units into a tree [MT88].

- *s*1: Montreal is a bilingual city.
- *s*2: They speak French and English.
- *s*3*b*: But the French sounds a little funny.

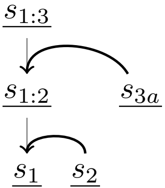


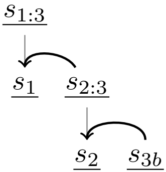


Rhetorical structure theory

In RST, discourse relations compose elementary units into a tree [MT88].

- *s*1: Montreal is a bilingual city.
- *s*2: They speak French and English.
- *s*3*b*: But the French sounds a little funny.
- Elementary discourse units (EDUs) $\approx$ clauses
- Relations include: cause-effect, comparison, temporal order, ...





RST Results

	Span	Nuclearity	Relation
Annotator agreement	88.7	77.7	65.8

Shift-reduce discourse parsing

Incremental (transition-based) parsing

- Keep a stack with elements $s_1, s_2, \ldots, s_N$
- The unread part of the discourse is on a queue, $q_0, \ldots q_M$
- At each step, make a decision whether to **shift** or **reduce**:

$$\hat{a} = \arg\max_{a} \boldsymbol{\theta}^{\top} \mathbf{f}(s_1, s_2, q_0, a)$$

Basic features

Sagae combined shift-reduce and perceptron, using many surface-level features [Sag09]:

- First/last words and POS, e.g., however
- Distance between discourse units
- "Head set": words with dependencies outside the unit (often main verbs)

More recent work

- **Feng and Hirst**: better features [FH12]
- **Joty et al**: more complex algorithm [JCNM13]

RST Results

	Span	Nuclearity	Relation
Annotator agreement	88.7	77.7	65.8

RST Results

	Span	Nuclearity	Relation
Annotator agreement	88.7	77.7	65.8
HILDA [HPdI10]	83.0	68.4	54.8
TSP [JCNM13]	82.7	68.4	55.7
“Basic features”	79.4	68.0	53.0

Why should these features work?

Our example:

- s_1: Montreal is a bilingual city.
- s_2: They speak French and English.
- q_0: This makes it interesting to visit.

Features:

s_1 first word: Montreal, head set: city, ...

s_2 first word: they, head set: speak, ...

q_0 first word: this, head set: makes, ...

Adding learned representations: first try

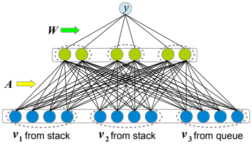
- Let's add a distributed representation for each discourse unit!
- Compositionality (see [LLH14])

inter-unit	"strong compositionality criterion" use the nucleus, ignore the satellite
intra-unit	just add up word representations for each word in the unit. (Blacoe and Lapata show that this is not as crazy as it sounds [BL12].)

- Word representations
  - Collobert and Weston [CW08]
  - Non-negative matrix factorization

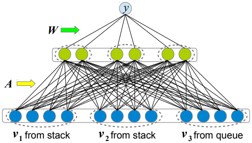
Representation schemes

$$\mathbf{f}(\mathbf{v}, \mathbf{A}) = \mathbf{A} \begin{bmatrix} \mathbf{v}_{s_1} \\ \mathbf{v}_{s_2} \\ \mathbf{v}_{q_1} \end{bmatrix}$$

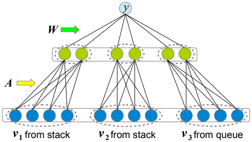


Representation schemes

$$\mathbf{f}(\mathbf{v}, \mathbf{A}) = \mathbf{A} \begin{bmatrix} \mathbf{v}_{s_1} \\ \mathbf{v}_{s_2} \\ \mathbf{v}_{q_1} \end{bmatrix}$$

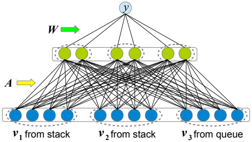


$$\mathbf{f}(\mathbf{v}, \mathbf{A}) = \begin{bmatrix} \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbf{v}_{s_1} \\ \mathbf{v}_{s_2} \\ \mathbf{v}_{q_1} \end{bmatrix}$$

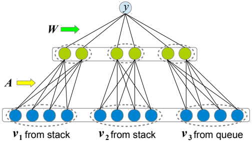


Representation schemes

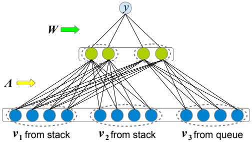
$$\mathbf{f}(\mathbf{v}, \mathbf{A}) = \mathbf{A} \begin{bmatrix} \mathbf{v}_{s_1} \\ \mathbf{v}_{s_2} \\ \mathbf{v}_{q_1} \end{bmatrix}$$



$$\mathbf{f}(\mathbf{v}, \mathbf{A}) = \begin{bmatrix} \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbf{v}_{s_1} \\ \mathbf{v}_{s_2} \\ \mathbf{v}_{q_1} \end{bmatrix}$$

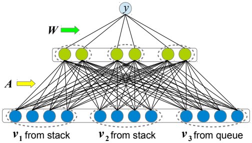


$$\mathbf{f}(\mathbf{v}, \mathbf{A}) = \begin{bmatrix} \mathbf{C} & -\mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{C} & -\mathbf{C} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v}_{s_1} \\ \mathbf{v}_{s_2} \\ \mathbf{v}_{q_1} \end{bmatrix}$$

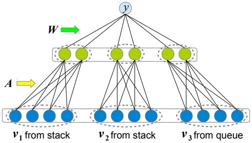


Representation schemes

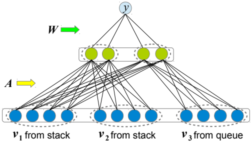
$$\mathbf{f}(\mathbf{v}, \mathbf{A}) = \mathbf{A} \begin{bmatrix} \mathbf{v}_{s_1} \\ \mathbf{v}_{s_2} \\ \mathbf{v}_{q_1} \end{bmatrix}$$



$$\mathbf{f}(\mathbf{v}, \mathbf{A}) = \begin{bmatrix} \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbf{v}_{s_1} \\ \mathbf{v}_{s_2} \\ \mathbf{v}_{q_1} \end{bmatrix}$$



$$\mathbf{f}(\mathbf{v}, \mathbf{A}) = \begin{bmatrix} \mathbf{C} & -\mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{C} & -\mathbf{C} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v}_{s_1} \\ \mathbf{v}_{s_2} \\ \mathbf{v}_{q_1} \end{bmatrix}$$



The concatenation form does best in most cases, but see the paper for details.

RST Results

	Span	Nuclearity	Relation
Annotator agreement	88.7	77.7	65.8
HILDA [HPdI10]	83.0	68.4	54.8
TSP [JCNM13]	82.7	68.4	55.7
“Basic features”	79.4	68.0	53.0

RST Results

	Span	Nuclearity	Relation
Annotator agreement	88.7	77.7	65.8
HILDA [HPdI10]	83.0	68.4	54.8
TSP [JCNM13]	82.7	68.4	55.7
"Basic features"	79.4	68.0	53.0
Word embeddings	75.3	67.1	53.8
NMF	78.6	67.7	54.8

Adding learned representations: second try

- Let's learn the word representations jointly with the parser!
- Basically, a hidden-variable support vector machine. Iterate:
  - Solve SVM dual objective
  - Perform gradient update to word representations

RST Results

	Span	Nuclearity	Relation
Annotator agreement	88.7	77.7	65.8
HILDA [HPdI10]	83.0	68.4	54.8
TSP [JCNM13]	82.7	68.4	55.7
"Basic features"	79.4	68.0	53.0
Word embeddings	75.3	67.1	53.8
NMF	78.6	67.7	54.8

RST Results

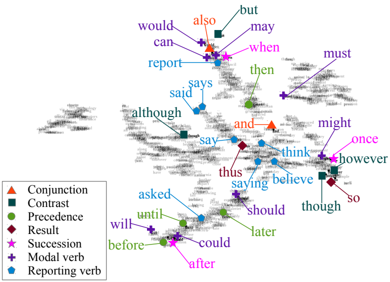
	Span	Nuclearity	Relation
Annotator agreement	88.7	77.7	65.8
HILDA [HPdI10]	83.0	68.4	54.8
TSP [JCNM13]	82.7	68.4	55.7
"Basic features"	79.4	68.0	53.0
Word embeddings	75.3	67.1	53.8
NMF	78.6	67.7	54.8
Representation learning	80.9	69.4	59.0

RST Results

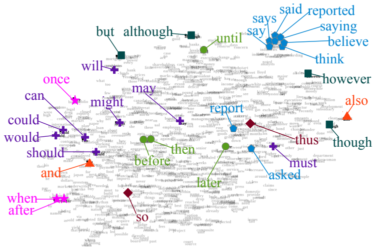
	Span	Nuclearity	Relation
Annotator agreement	88.7	77.7	65.8
HILDA [HPdI10]	**83.0**	68.4	54.8
TSP [JCNM13]	82.7	68.4	55.7
"Basic features"	79.4	68.0	53.0
Word embeddings	75.3	67.1	53.8
NMF	78.6	67.7	54.8
Representation learning	80.9	69.4	59.0
+basic features	82.1	**71.1**	**61.6**

On discourse relations, representation learning cuts the gap between SOTA and inter-annotator agreement by 60%!

Representation learned



NMF, $K = 20$



Representation learning, $K = 20$

This talk

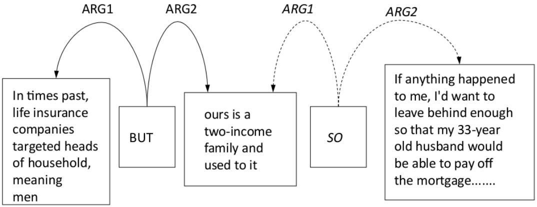
Representation learning for two discourse structures:

- **Rhetorical structure theory**: learn to parse discourse into trees, while jointly learning word representations [JE14b].
- **Penn Discourse Treebank**: learn compositional operators for distributed semantics [JE14a].

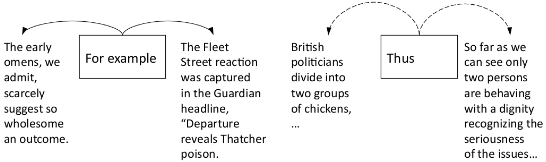
Yangfeng Ji

Penn Discourse Treebank

- Spans can participate in multiple relations



- Relations need not link up to cover the text



Implicit relation classification

- Implicit discourse connectives are annotated:
  - Bob gave Tina the burger
  - (Because) she was hungry

  There are 16 classes of level-2 connectives
- Existing approaches again emphasize bilexical features [LKN09].
- Sparsity is again a problem: $\langle$burger, hungry$\rangle$, $\langle$knish, hungry$\rangle$, $\langle$poutine, hungry$\rangle$, ...

A bilinear model

$$\hat{y} = \underset{y \in \mathcal{Y}}{\arg\max} \quad (\mathbf{u}^{(\ell)})^{\mathsf{T}} \mathbf{A}_y \mathbf{u}^{(r)} + b_y$$

- $y \in \mathcal{Y}$ is a relation
- $\mathbf{u}^{(\ell)}$ is the representation of the left argument
- $\mathbf{u}^{(r)}$ is the representation of the right argument
- In practice, we set

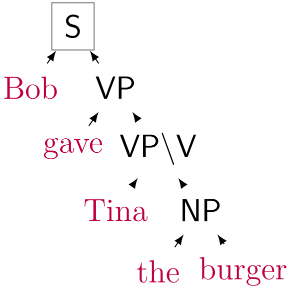
$$\mathbf{A}_y = \mathbf{a}_{y,1}\mathbf{a}_{y,2}^{\mathsf{T}} + \mathrm{diag}(\mathbf{a}_{y,3})$$

to avoid overfitting.

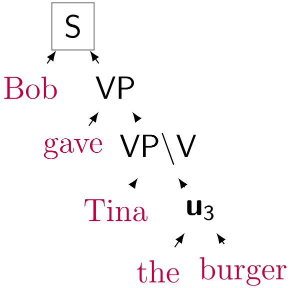
PDTB Results

Most common class	26.0
Additive word representations	**28.7**
Lin et al [LKN09]	40.2
Our reimplementation of Lin et al	39.7

Vector-semantic composition

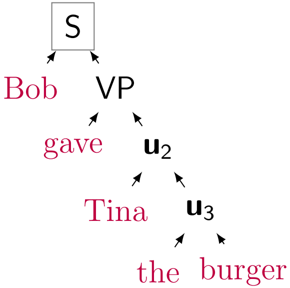


Vector-semantic composition



$$\mathbf{u}_3 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{the}}^{\mathsf{T}}\ \mathbf{u}_{\text{burger}}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

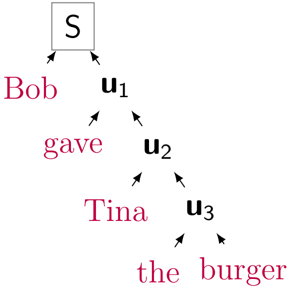
Vector-semantic composition



$$\mathbf{u}_3 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{the}}^{T}\ \mathbf{u}_{\text{burger}}^{T}\right]^{T}\right)$$

$$\mathbf{u}_2 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{Tina}}^{T}\ \mathbf{u}_{3}^{T}\right]^{T}\right)$$

Vector-semantic composition

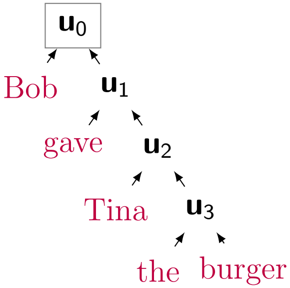


$$\mathbf{u}_3 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{the}}^{\mathsf{T}}\ \mathbf{u}_{\text{burger}}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

$$\mathbf{u}_2 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{Tina}}^{\mathsf{T}}\ \mathbf{u}_{3}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

$$\mathbf{u}_1 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{gave}}^{\mathsf{T}}\ \mathbf{u}_{2}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

Vector-semantic composition



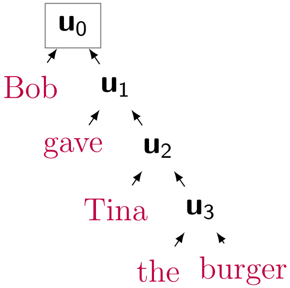
$$\mathbf{u}_3 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{the}}^{\mathsf{T}}\ \mathbf{u}_{\text{burger}}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

$$\mathbf{u}_2 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{Tina}}^{\mathsf{T}}\ \mathbf{u}_{3}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

$$\mathbf{u}_1 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{gave}}^{\mathsf{T}}\ \mathbf{u}_{2}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

$$\mathbf{u}_0 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{Bob}}^{\mathsf{T}}\ \mathbf{u}_{1}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

Vector-semantic composition



$$\mathbf{u}_3 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{the}}^{\mathsf{T}}\ \mathbf{u}_{\text{burger}}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

$$\mathbf{u}_2 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{Tina}}^{\mathsf{T}}\ \mathbf{u}_{3}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

$$\mathbf{u}_1 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{gave}}^{\mathsf{T}}\ \mathbf{u}_{2}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

$$\mathbf{u}_0 = \tanh\left(\mathbf{U}\left[\mathbf{u}_{\text{Bob}}^{\mathsf{T}}\ \mathbf{u}_{1}^{\mathsf{T}}\right]^{\mathsf{T}}\right)$$

- DISCO2: **Dis**tributional **co**mpositional semantics for **disco**urse.
- Same architecure as Socher et al [SHP+11].
- The matrix **U** is learned by backpropagating from a hinge loss on relation classification.

PDTB Results

Most common class	26.0
Additive word representations	**28.7**
Lin et al [LKN09]	40.2
Our reimplementation of Lin et al	39.7

PDTB Results

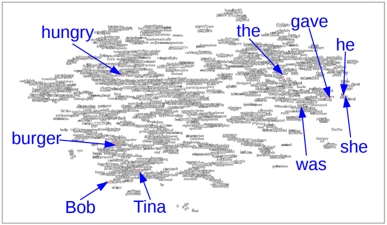
Most common class	26.0
Additive word representations	28.7
Lin et al [LKN09]	40.2
Our reimplementation of Lin et al	39.7
Disco2	37.0
Disco2 + [LKN09] features	42.5

Are we done?

- Bob gave Tina the burger.
- **She** was hungry.

- Bob gave Tina the burger.
- **He** was hungry.

The discourse relations are completely different.
The distributed representations are nearly identical.



One vector is not enough.

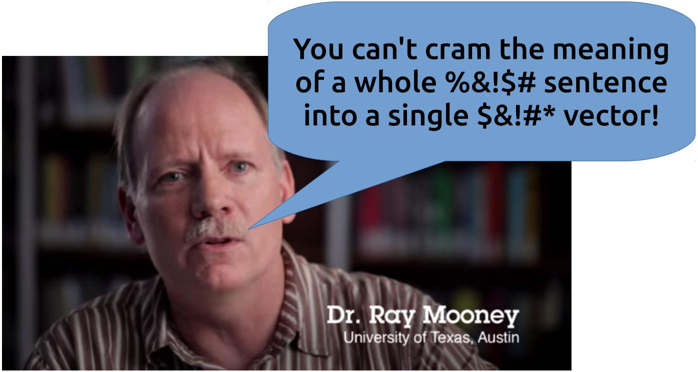
If we insist on representing each discourse argument as a single vector, we lose the ability to track references across the discourse.

Or to put it another way...

One vector is not enough.

If we insist on representing each discourse argument as a single vector, we lose the ability to track references across the discourse.

Or to put it another way...



Entity-augmented distributed semantics

Look at things from Tina's perspective:

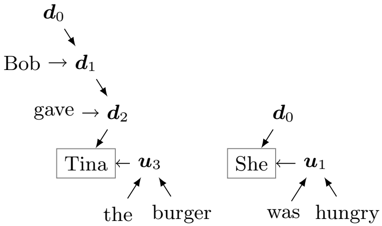
- $s1$: She got the burger from Bob
- $s2$: She was hungry

Let's represent these Tina-centric meanings with more vectors!

The downward pass

A **downward pass** computes a downward vector for each node in the parse.

$$\mathbf{d}_i = \tanh\left(\mathbf{V}\begin{bmatrix}\mathbf{d}_{\rho(i)}\\ \mathbf{u}_{s(i)}\end{bmatrix}\right)$$



This computation preserves the feedforward architecture.

A new bilinear model

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} (\mathbf{u}^{(\ell)})^\top \mathbf{A}_y \mathbf{u}^{(r)} + \sum_{\langle i,j \rangle \in \mathcal{A}} (\mathbf{d}_i^{(\ell)})^\top \mathbf{B}_y \mathbf{d}_j^{(r)} + b_y$$

We now sum over coreferent mention pairs $\langle i, j \rangle \in \mathcal{A}$, obtained from the Berkeley coreference system.

PDTB Results

Most common class	26.0
Additive word representations	28.7
Lin et al [LKN09]	40.2
Our reimplementation of Lin et al	39.7
Disco2	37.0
Disco2 + [LKN09] features	42.5

PDTB Results

Most common class	26.0
Additive word representations	28.7
Lin et al [LKN09]	40.2
Our reimplementation of Lin et al	39.7
Disco2	37.0
Disco2 + [LKN09] features	42.5
Disco2 + [LKN09] features + entity semantics	**43.6**

PDTB Results

Most common class	26.0
Additive word representations	28.7
Lin et al [LKN09]	40.2
Our reimplementation of Lin et al	39.7
Disco2	37.0
Disco2 + [LKN09] features	42.5
Disco2 + [LKN09] features + entity semantics	**43.6**

- Only 30% of PDTB relation pairs have coreferent mentions (according to Berkeley coref).
- On these examples, the improvement is 2.7%.

Between vectors and lambdas

- Pure vector semantics are insufficiently expressive for discourse analysis.
- But broad-coverage formal semantic parsing is (currently) too brittle.
- What's in between?
  - Lewis and Steedman: make formal semantics a little more distributional [LS13].
  - Entity-augmented distributed semantics: make distributed semantics a little more formal.

Summary: a call to arms

- Discourse relations are all about **meaning**, and black-box machine learning won't work.
  - Simple surface features are insufficiently expressive to capture semantics.
  - More complex surface features cause overfitting.
- Discourse also connects with a huge number of NLP applications.
- Machine learning researchers: **join the fight!**

William Blacoe and Mirella Lapata.
A comparison of vector-based representations for semantic composition.
In *Proceedings of Empirical Methods for Natural Language Processing (EMNLP)*, pages 546–556, 2012.

Ronan Collobert and Jason Weston.
A unified architecture for natural language processing: Deep neural networks with multitask learning.
In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 160–167, 2008.

Vanessa Wei Feng and Graeme Hirst.
Text-level Discourse Parsing with Rich Linguistic Features.
In *ACL*, 2012.

Hugo Hernault, Helmut Prendinger, David A. duVerle, and Mitsuru Ishizuka.
HILDA: A Discourse Parser Using Support Vector Machine Classification.
Dialogue and Discourse, 1(3):1–33, 2010.

Shafiq Joty, Giuseppe Carenini, Raymond Ng, and Yashar Mehdad.
Combining Intra- and Multi-sentential Rhetorical Parsing for Document-level Discourse Analysis.
In *ACL*, 2013.

Yangfeng Ji and Jacob Eisenstein.
One vector is not enough: Entity-augmented distributional semantics for discourse relations.
Technical Report http://arxiv.org/abs/1411.6699, 2014.

Yangfeng Ji and Jacob Eisenstein.
Representation learning for text-level discourse parsing.
In *Proceedings of the Association for Computational Linguistics (ACL)*, Baltimore, MD, 2014.

Ziheng Lin, Min-Yen Kan, and Hwee Tou Ng.
Recognizing implicit discourse relations in the penn discourse treebank.
In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 343–351. Association for Computational Linguistics, 2009.

Jiwei Li, Rumeng Li, and Eduard Hovy.
Recursive deep models for discourse parsing.
In *Proceedings of Empirical Methods for Natural Language Processing (EMNLP)*, 2014.

Mike Lewis and Mark Steedman.
Combined distributional and logical semantics.
Transactions of the Association for Computational Linguistics, 1:179–192, 2013.

William C Mann and Sandra A Thompson.
Rhetorical structure theory: Toward a functional theory of text organization.
Text, 8(3):243–281, 1988.

Kenji Sagae.
Analysis of Discourse Structure with Syntactic Dependencies and Data-Driven Shift-Reduce Parsing.
In *Proceedings of the 11th International Conference on Parsing Technologies (IWPT'09)*, pages 81–84, Paris, France, October 2009. Association for Computational Linguistics.

Richard Socher, Eric H. Huang, Jeffrey Pennington, Andrew Y. Ng, and Christopher D. Manning.
Dynamic Pooling And Unfolding Recursive Autoencoders For Paraphrase Detection.
In *Advances in Neural Information Processing Systems (NIPS)*, 2011.

Kimberly Voll and Maite Taboada.
Not all words are created equal: Extracting semantic orientation as a function of adjective relevance.
In *Proceedings of Australian Conference on Artificial Intelligence*, 2007.