Compositional Models of Vector-based Semantics: From Theory to Tractable Implementation

Day 5: Evaluation. Experimenting with Semantic Tasks

Gijs Wijnholds & Michael Moortgat

ESSLLI 2022

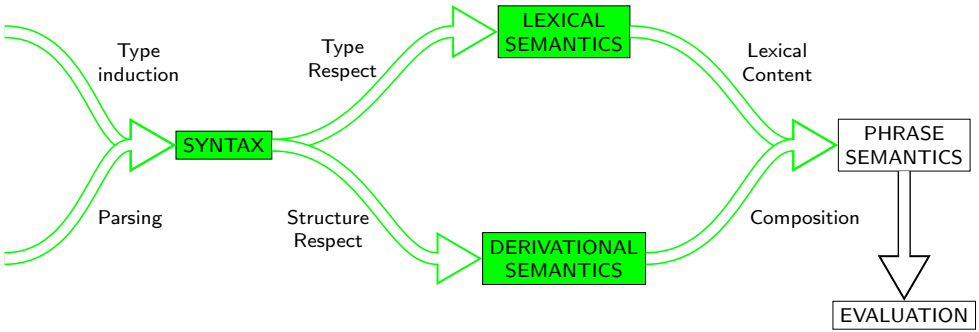
Abstract

Vector-based compositional architectures combine a distributional view of word meanings with a modelling of the syntax-semantics interface as a structure-preserving map relating syntactic categories (types) and derivations to their counterparts in a corresponding meaning algebra.

This design is theoretically attractive, but faces challenges when it comes to large-scale practical applications. First there is the curse of dimensionality resulting from the fact that semantic spaces directly reflect the complexity of the types of the syntactic front end. Secondly, modelling of the meaning algebra in terms of finite dimensional vector spaces and linear maps means that vital information encoded in syntactic derivations is lost in translation.

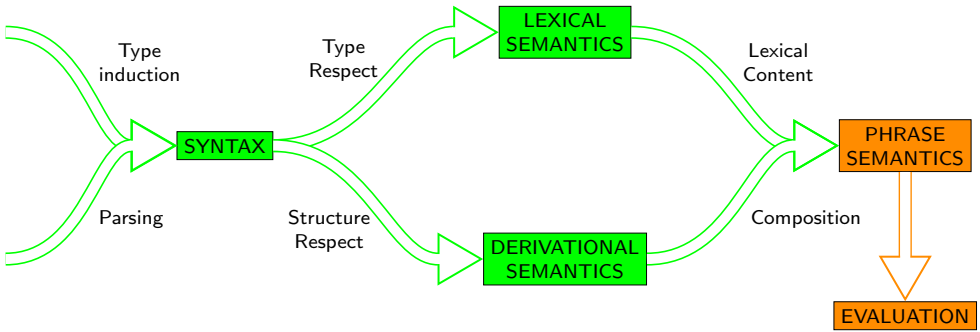
The course compares and evaluates methods that are being proposed to face these challenges. Participants gain a thorough understanding of theoretical and practical issues involved, and acquire hands-on experience with a set of user-friendly tools and resources.

Recap: The Compositional Process



- Our core methodology provides syntax and the interfacing with semantics,
- Lexical content is learnable, though not always in a tractable way (Tue)
- Syntax doesn't come for free: type induction & parsing as learnable processes (Wed/Thu)
- In the end, the phrase semantics can be applied to NLP tasks (Fri)

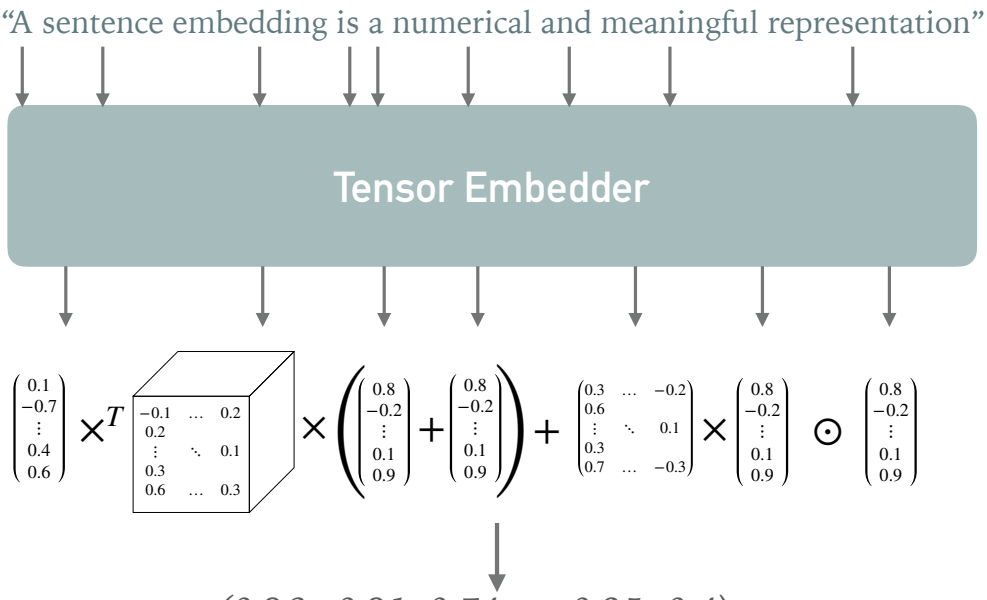
Today: The Compositional Process



- Evaluating tensor-based models
- Linguistic variation: Dutch vs English Natural Language Inference
- Discontinuous constituency and lexical knowledge: taking down BERT
- Closing off, and what is next?

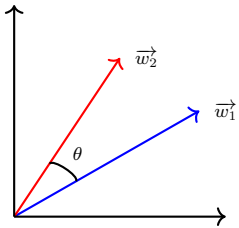
Evaluating composition models

Dream Machine



Assessing Word Similarity

Comparing Vectors Recall that we can use cosine similarity between vectors:



$$\text{sim}(\overrightarrow{w_1}, \overrightarrow{w_2}) = \cos(\theta) = \frac{\overrightarrow{w_1} \cdot \overrightarrow{w_2}}{|\overrightarrow{w_1}||\overrightarrow{w_2}|}$$

Word Similarity Datasets By asking participants to rank word pairs for similarity on a scale (e.g. 1-7), taking the average rating, we get gold standard similarity ratings.

Evaluating Models To get to a single score for a given word embedding model, we compute the Spearman ρ rank correlation coefficient of the gold standard similarity against the cosine similarity over vectors for the word pairs in the dataset. Spearman ρ values lie between 1 (ranking by gold standard equals ranking by cosine similarity) and -1 (ranking by gold standard is reverse w.r.t. ranking by cosine similarity)

A Brief History of Word Similarity Datasets

Word similarity datasets

Name	#Pairs	Categories	Reference
RG	65	nouns	Rubenstein and Goodenough [1965]
MC30	30	nouns	Miller and Charles [1991]
WordSim353	353	nouns	Finkelstein et al. [2001]
VerbSim	130	verbs	Yang and Powers [2006]
MEN	3000	nouns, adjectives, verbs	Bruni et al. [2012]
SimLex	999	nouns, adjectives, verbs	Hill et al. [2015]
SimVerb	3500	verbs	Gerz et al. [2016]

Word similarity results

	RG	WordSim353	MC30	SimLex	MEN
Count	0.608	0.358	0.546	0.259	0.553
Word2Vec	0.823	**0.698**	**0.768**	**0.403**	**0.781**
GloVe	**0.831**	0.618	0.738	0.390	0.773
FastText	0.772	0.546	0.696	0.402	0.768

Evaluating Tensors

No Tensor in Cosine For matrices (order 2), cubes (order 3), tesseracts (order 4) etc. cosine similarity doesn't apply (viz. what is the angle between two matrices?)

Parametric Comparison We view tensors as maps: two tensors are similar when they transform the same arguments into similar vectors:

$$\text{tensorsim}(\mathbf{T_1}, \mathbf{T_2}) = \underset{\langle d_1,\ldots,d_n\rangle \in \mathcal{D}}{\text{med}} \cos(\mathbf{T_1}\overrightarrow{d_1}\ldots\overrightarrow{d_n}, \mathbf{T_2}\overrightarrow{d_1}\ldots\overrightarrow{d_n})$$

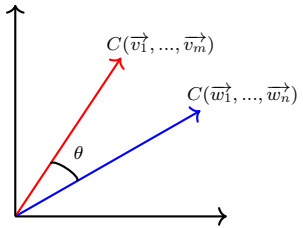
New Formulas For two verbs V_1, V_2:

$$\begin{aligned}
\text{matsim}^S &\quad \underset{\overrightarrow{s}\in\mathcal{S}}{\text{med}} \cos(\overline{V_1}\overrightarrow{s}, \overline{V_2}\overrightarrow{s}) \\
\text{matsim}^O &\quad \underset{\overrightarrow{o}\in\mathcal{O}}{\text{med}} \cos(\overline{V_1}\overrightarrow{o}, \overline{V_2}\overrightarrow{o}) \\
\text{cubesim} &\quad \underset{\langle\overrightarrow{s},\overrightarrow{o}\rangle\in\mathcal{A}}{\text{med}} \cos(\mathbf{V_1}\overrightarrow{o}\overrightarrow{s}, \mathbf{V_2}\overrightarrow{o}\overrightarrow{s})
\end{aligned}$$

Combinatorial Issues It is intractable to consider all possible word vectors in the vocabulary, so instead we can take the centroids of a bunch of word vectors as arguments to the tensors that we compare.

Assessing Sentence Similarity

Cosine Everywhere Cosine similarity naturally extends to sentence vectors:



$$\text{sim}(C(\overrightarrow{w_1}, ..., \overrightarrow{w_n}), C(\overrightarrow{v_1}, ..., \overrightarrow{v_m})) =$$

$$\cos(\theta) = \frac{C(\overrightarrow{w_1}, ..., \overrightarrow{w_n}) \cdot C(\overrightarrow{v_1}, ..., \overrightarrow{v_m})}{|C(\overrightarrow{w_1}, ..., \overrightarrow{w_n})||C(\overrightarrow{v_1}, ..., \overrightarrow{v_m})|}$$

Disambiguation in context Given a verb with two interpretations, and some context, choose the correct interpretation:

painter draw sword	painter pull sword	painter depict sword
samurai draw sword	samurai pull sword	samurai depict sword

Comparing sentences Given two sentences, how similar are they?

painter draw sword vs. author write book

A Brief History of Evaluation: Sentences

Sentence similarity datasets

Name	#Pairs	Sentences	Task	Reference
ML2008	200	$s\ v/v\ o$	disambiguation	Mitchell and Lapata [2008]
ML2010	200	$s\ v/v\ o$	similarity	Mitchell and Lapata [2010]
GS2011	200	$s\ v\ o$	disambiguation	Grefenstette and Sadrzadeh [2011]
KS2013a	100	$s\ v\ o$	disambiguation	Kartsaklis and Sadrzadeh [2013]
KS2013b	108	$s\ v\ o$	similarity	Kartsaklis et al. [2013]
ELLDIS	400	$s\ v\ o\ and\ s^*\ too$	disambiguation	Wijnholds and Sadrzadeh [2019]
ELLSIM	432	$s\ v\ o\ and\ s^*\ too$	similarity	Wijnholds and Sadrzadeh [2019]

The Task Given a sentence, decide which vectors and tensors, in combination with which composition model, gives the best correlation scores.

A Remark These datasets target sentences in a specific format, so they are limited in scope, but allow us to evaluate very specifically*

*there are many other, more general sentence level datasets, to be discussed a bit later today

Composition Models

Transitive sentences [Milajevs et al., 2014] performs an evaluation study for several datasets containing transitive sentences, and gathers several composition models.

Verb matrix Two ways of constructing a verb matrix:

$$\overline{verb} = \sum_{ij} \overrightarrow{subj}_i \otimes \overrightarrow{obj}_j \qquad \widetilde{verb} = \overrightarrow{verb} \otimes \overrightarrow{verb}$$

Composition models

Verb only	$\overrightarrow{verb}/\overline{verb}/\widetilde{verb}$
Additive	$\overrightarrow{subj} + \overrightarrow{verb} + \overrightarrow{obj}$
Multiplicative	$\overrightarrow{subj} \odot \overrightarrow{verb} \odot \overrightarrow{obj}$
Relational	$\overline{verb} \odot (\overrightarrow{subj} \otimes \overrightarrow{obj})$
Kronecker	$(\overrightarrow{verb} \otimes \overrightarrow{verb}) \odot (\overrightarrow{subj} \otimes \overrightarrow{obj})$
Copy Subject	$\overrightarrow{subj} \odot (\overline{verb} \times \overrightarrow{obj})$
Copy Object	$\overrightarrow{obj} \odot (\overline{verb}^T \times \overrightarrow{subj})$
Frobenius add.	$(\overrightarrow{subj} \odot (\overline{verb} \times \overrightarrow{obj})) + (\overrightarrow{obj} \odot (\overline{verb}^T \times \overrightarrow{subj}))$
Frobenius mult.	$(\overrightarrow{subj} \odot (\overline{verb} \times \overrightarrow{obj})) \odot (\overrightarrow{obj} \odot (\overline{verb}^T \times \overrightarrow{subj}))$
Frobenius outer	$(\overrightarrow{subj} \odot (\overline{verb} \times \overrightarrow{obj})) \otimes (\overrightarrow{obj} \odot (\overline{verb}^T \times \overrightarrow{subj}))$

Sentence dataset results

Disambiguation Tensor-based for the win! :-) [Wijnholds, 2020]

KS2013a	Count Based		Word2Vec		GloVe		FastText	
Verb Only Vector	0.108		0.199		0.132		0.112	
Verb Only Tensor	0.093		0.100		0.065		0.040	
Additive	0.104		0.210		0.174		0.117	
Multiplicative	0.279		0.334		0.110		0.302	
Best Tensor	0.322	**FM** $\widetilde{\cdot}$	**0.415**	**FA** $\widetilde{\cdot}$	0.140	**FA** $\widetilde{\cdot}$	0.368	**FA** $\widetilde{\cdot}$
2nd Best Tensor	0.258	**CO** $\widetilde{\cdot}$	**0.408**	**CS** $\widetilde{\cdot}$	0.123	**FO** $\widetilde{\cdot}$	0.334	**CS** $\widetilde{\cdot}$

Similarity No way to beat addition :-(

KS2013b	Count Based		Word2Vec		GloVe		FastText	
Verb Only Vector	0.521		0.665		0.535		0.705	
Verb Only Tensor	0.456		0.617		0.504		0.563	
Additive	0.677		**0.763**		0.719		0.764	
Multiplicative	0.719		0.528		0.283		0.587	
Best Tensor	**0.745**	**FM** $\widetilde{\cdot}$	0.623	**RE** $\bar{\cdot}$	0.427	**FO** $\bar{\cdot}$	0.641	**FO** $\bar{\cdot}$
2nd Best Tensor	0.739	**FO** $\widetilde{\cdot}$	0.578	**RE** $\widetilde{\cdot}$	0.418	**RE** $\bar{\cdot}$	0.637	**RE** $\bar{\cdot}$

Composing Elliptical Phrases

To resolve or not to resolve For a sentence [Wijnholds and Sadrzadeh, 2019]

$$\texttt{subj verb obj and subj}^* \texttt{ does too}$$

we can use the transitive composition models from before (denoted by T) and now distinguish between models that resolve ellipsis or not:

Composition Models

Model	**Formula** ($\star$: addition, multiplication)
Verb only	$\overrightarrow{verb}/\overline{verb}/\widetilde{verb}$
Additive	$\overrightarrow{subj} + \overrightarrow{verb} + \overrightarrow{obj} + \overrightarrow{and} + \overrightarrow{subj^*} + \overrightarrow{does} + \overrightarrow{too}$
Multiplicative	$\overrightarrow{subj} \odot \overrightarrow{verb} \odot \overrightarrow{obj} \odot \overrightarrow{and} \odot \overrightarrow{subj^*} \odot \overrightarrow{does} \odot \overrightarrow{too}$
Additive non-linear	$\overrightarrow{subj} + \overrightarrow{verb} + \overrightarrow{obj} + \overrightarrow{subj^*} + \overrightarrow{verb} + \overrightarrow{obj}$
Multiplicative non-linear	$\overrightarrow{subj} \odot \overrightarrow{verb} \odot \overrightarrow{obj} \odot \overrightarrow{subj^*} \odot \overrightarrow{verb} \odot \overrightarrow{obj}$
Tensor Based	$T(\overrightarrow{subj}, \overline{verb}, \overrightarrow{obj}) \star T(\overrightarrow{subj^*}, \overline{verb}, \overrightarrow{obj})$
Frobenius	$T(\overrightarrow{subj} \star \overrightarrow{subj^*}, \overline{verb}, \overrightarrow{obj})$

Ellipsis Disambiguation Results

ELLDIS	Count Based		Word2Vec		GloVe		FastText	
Verb Only Vector	0.436		0.241		0.445		0.229	
Verb Only Tensor	0.330		0.438		0.394		0.388	
Add. Linear	0.442		0.273		0.305		0.141	
Mult. Linear	0.325		-0.012		0.182		0.293	
Add. Non-Linear	0.445		0.328		0.326		0.140	
Mult. Non-Linear	0.503		0.209		0.245		0.044	
Best Tensor	**0.539**	**CO** $\widetilde{+}$	0.462	**FA** $\overline{+}$	0.373	**COS** $\overline{+}$	0.494	**FO** $\overline{+}$
2nd Best Tensor	0.526	**FA** $\widetilde{+}$	0.454	**FO** $\overline{+}$	0.369	**FA** $\overline{+}$	0.465	**FA** $\overline{+}$
Best KaCo	**0.539**	**CO** $\widetilde{\odot}$	0.462	**FA** $\overline{+}$	0.397	**FA** $\overline{\odot}$	0.497	**FO** $\overline{+}$
2nd Best KaCo	**0.527**	**FO** $\widetilde{\odot}$	0.460	**FO** $\overline{+}$	0.369	**FA** $\overline{+}$	0.465	**FA** $\overline{+}$

Analysis

- It pays to resolve ellipsis for the baseline models
- Tensor-based better, no distinction between classical and Frobenius copying.

Ellipsis Similarity Results

ELLSIM	Count Based		Word2Vec		GloVe		FastText	
Verb Only Vector	0.457		0.583		0.435		0.647	
Verb Only Tensor	0.395		0.566		0.443		0.534	
Add. Linear	0.700		0.726		0.696		0.741	
Mult. Linear	0.633		0.130		0.367		0.199	
Add. Non-Linear	0.681		**0.762**		0.710		0.739	
Mult. Non-Linear	0.723		0.355		0.244		0.450	
Best Tensor	**0.741**	**FO** $\widetilde{\odot}$	0.706	**RE** $\overline{+}$	0.491	**FA** $\overline{+}$	0.699	**FO** $\overline{+}$
2nd Best Tensor	0.737	**FA** $\widetilde{\odot}$	0.671	**FO** $\overline{+}$	0.482	**FO** $\overline{+}$	0.688	**CO** $\overline{\odot}$
Best KaCo	0.732	**FM** $\widetilde{\odot}$	0.706	**RE** $\overline{+}$	0.491	**FA** $\overline{+}$	0.702	**FO** $\overline{+}$
2nd Best KaCo	0.730	**FA** $\widetilde{\odot}$	0.668	**FO** $\overline{+}$	0.486	**FO** $\overline{+}$	0.682	**RE** $\overline{+}$

Analysis

- It pays to resolve ellipsis (again)
- Tensor-based may outperform linear addition
- Classical copying beats Frobenius copying

Neural Verb Tensors to the Rescue?

Tensor Skipgram Remember that we could learn 'decomposed' tensor representations for verbs:

Representation	Rank	Context
$\overrightarrow{v}_a$	vector	linear window
$\overrightarrow{v}_s / \overrightarrow{v}_o / \overrightarrow{v}_b$	vector	objects/subjects/both
$\overline{V}_a^S / \overline{V}_a^O$	matrix	full sentence
$\overline{V}_o^S / \overline{V}_s^O$	matrix	objects/subjects
$\overline{\overline{V}}_a$	cube	full sentence

Verb Similarity

	$\overrightarrow{v}_a$	$\overrightarrow{v}_{s/o/b}$	$\overline{V}_a$	$\overline{V}_{s/o}$	$\mathbb{V}_a$
MEN$_v$	0.282	0.248	0.500	**0.589**	0.035
SimLex$_v$	0.046	0.272	0.163	**0.340**	0.024
VerbSim	0.338	**0.563**	0.085	0.550	-0.076
SimVerb$_d$	0.224	0.249	-0.023	**0.291**	-0.012
SimVerb$_t$	0.183	0.197	0.019	**0.240**	-0.025

Neural Verb Tensors vs. Sesame Street

Results [Wijnholds et al., 2020] compares verb skipgram tensors against other tensor-based models, and to state-of-the-art sentence embedding methods:

	ML08	ML10	GS11	KS13a	KS13b
$C(+)$	0.17	0.54	0.19	0.18	0.67
$C(\mathbf{V}_{Kron})$	0.08	0.40	0.20	0.28	0.53
$C(\mathbf{V}_{Rel})$	0.19	0.51	0.32	0.19	0.51
$C(\mathbb{V}_{(1)}^{sent\|s/o})$	-0.04	0.00	0.25	0.20	0.54
$C(\widetilde{\mathbb{V}}_{(1)}^{o\|s/s\|o})$	**0.19**	**0.55**	**0.54**	**0.37**	**0.75**
$C(\mathbb{V}_{(2)}^{sent\|s,o})$	—	—	-0.02	-0.04	0.06
Human	0.66	0.71	0.74	0.58	0.75

	ML08	ML10	GS11	KS13a	KS13b
$\widetilde{\mathbb{V}}_{(1)}^{o\|s/s\|o}$	0.19	0.55	0.54	**0.37**	0.75
IS	0.18	0.63	0.30	0.17	0.78
USE	0.04	0.33	0.09	0.21	0.54
ELMo	0.17	0.54	0.11	0.24	0.73
BERT$_p$	0.19	0.34	0.24	0.32	0.61
BERT$_f$	**0.32**	**0.74**	**0.61**	0.32	**0.82**
Human	0.66	0.71	0.74	0.58	0.75

More results Including the SICK dataset, which contains longer & 'natural' sentences.

	Add	Kron	Rel	$\widetilde{\mathbb{V}}_{(1)}^{o\|s/s\|o}$	IS	USE	BERT$_p$	BERT$_f$
ELLDIS	0.31	0.30	0.37	0.56	0.34	0.27	0.36	**0.65**
ELLSIM	0.67	0.52	0.65	0.76	**0.80**	0.68	0.67	0.79
SICK-R	0.71	0.58	0.44	0.70	0.74	**0.76**	0.70	**0.76**

What we learn from this

Disambiguation Tensor-based composition models seem to be better at disambiguation in context.

Similarity Simple addition of word vectors outperforms tensor-based models in the case of sentence similarity.

Composition Models vs Encoders The same pattern persists when we compare with sentence encoders, that can achieve higher results on sentence similarity but are still stuck at verb disambiguation in context.

To resolve or not to resolve In almost all cases, models resolve ellipsis achieve higher performance than models that don't. For the case of tensor-based models the resolution is handled by the syntactic front-end, whereas sentence encoders would require such a front-end to be implemented still $\rightarrow$ we need grammar(/semantics)!

The Great Disclaimer It is intractable to try out all possible representation methods, composition models, etc., and they do not easily scale up to larger evaluation datasets where sentence length is not given in advance. Even on the SICK dataset, compromises have to be made in order to evaluate these tensor representations.

Large-scale NLP Evaluation

Natural Language Inference

Entailment? Given a premise s_1, and a hypothesis s_2, decide whether the premise Entails, Contradicts, or is Neutral with respect to the hypothesis.

Human Inferences Some examples from the SICK [Marelli et al., 2014] dataset:

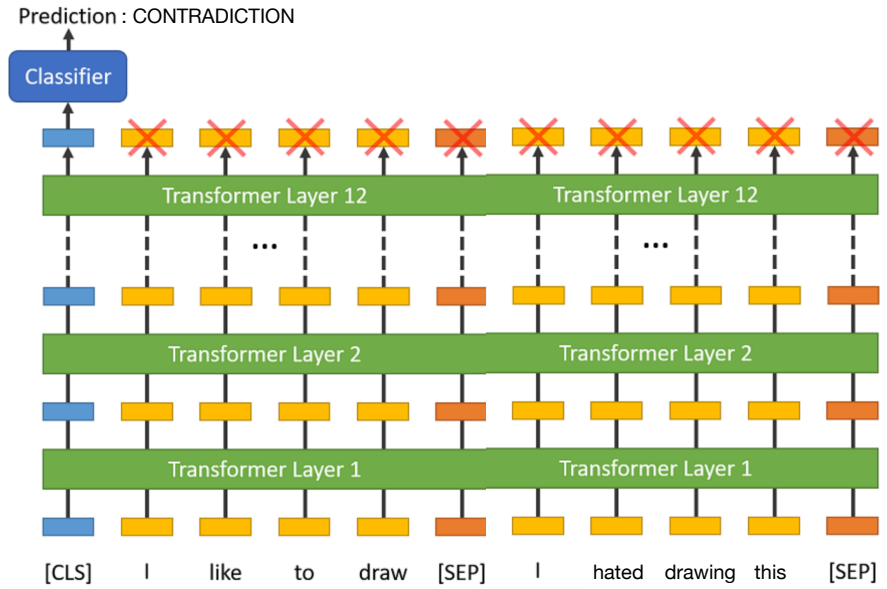
Premise	Hypothesis	Label
A deer is jumping over a fence	A deer isn't jumping over a fence	C
A player is running with the ball	Two teams are competing in a football match	N
An old man is sitting in a field	A man is sitting in a field	E

NLI Datasets

- Larger and more fuzzy: SNLI [Bowman et al., 2015],
- Multi-genre: MNLI [Williams et al., 2018],
- Multi-lingual: XNLI [Conneau et al., 2018],
- With explanations: e-SNLI [Camburu et al., 2018],
- etc...

BERT for NLI

Freeze! We can choose to fine-tune or to extract, by 'freezing' the BERT layers:



Analyzing NLI Models

Generalization Models trained on one NLI dataset may not perform well on another set [Talman and Chatzikyriakidis, 2019]:

Train	Test	Acc.	Model
SNLI	SNLI	90.4	BERT-base
SNLI	MNLI	75.5	BERT-base
SNLI	SICK	56.9	BERT-base
⋮	⋮	⋮	⋮

Specialized Reasoning Monotonicity Entailment Dataset [Yanaka et al., 2019]

Example:

P: *Every* [NP *person* ↓] [VP *bought a movie ticket* ↑]
H: *Every young person bought a ticket*

Model	Train	Upward	Downward	Non	All
BERT	SNLI	50.1	46.8	7.5	45.8

Analyzing/Probing Lots of work into investigating NLI models (and contextualized representations) [Naik et al., 2018, McCoy et al., 2019, Richardson et al., 2020, Tenney et al., 2019]

Reasoning in Dutch

SICK-NL [Wijnholds and Moortgat, 2021] semi-automatically translated the SICK dataset into Dutch.

Dutch vs English

	SICK	SICK-NL
No. of tokens	189783	176509
No. of unique tokens	2328	2870
Avg. sentence length	9.64	8.97
Avg. word overlap	66.91%	58.99%

Comparing BERTs A comparison shows that the Dutch incarnation is harder to solve:

SICK		SICK-NL	
BERT	87.34	BERTje	83.94
mBERT	87.02	mBERT	84.53
RoBERTa	90.11	RobBERT	82.02

Error Analysis

BERT		Prediction EN-NL				Prediction EN				Prediction NL			
		C	N	E	*rec.*	C	N	E	*rec.*	C	N	E	*rec.*
Gold	C	549	62	16	88%	46	27	12	54%	22	52	11	26%
	N	35	2341	147	93%	37	147	83	55%	32	116	119	43%
	E	3	143	1035	88%	3	53	167	75%	3	165	55	25%
	pr.	94%	92%	86%		53%	65%	64%		39%	35%	30%	
mBERT													
Gold	C	553	69	10	88%	24	47	9	30%	42	31	7	52%
	N	32	2344	106	94%	9	210	89	68%	71	93	144	30%
	E	3	160	1015	86%	0	103	123	54%	15	111	100	44%
	pr.	94%	91%	90%		73%	58%	56%		33%	40%	40%	
RoBERTa													
Gold	C	563	68	4	89%	27	46	4	35%	35	28	14	45%
	N	25	2301	82	96%	6	279	97	73%	79	96	207	25%
	E	1	108	995	90%	2	42	256	85%	15	246	39	13%
	pr.	96%	93%	92%		77%	76%	72%		27%	26%	15%	

Model mistakes Where the models disagree between the English/Dutch premise-hypothesis pairs, the Dutch models have a higher tendency to classify Entailment as Neutral, and Neutral as Entailment.

Patterns in Dutch

Prepositional phrase the attachment of a PP in Dutch may be interchangedly placed before or after the verb:

A woman is wakeboarding on a lake	→	Een vrouw is aan het wakeboarden op een meer
A woman is wakeboarding on a lake	→	Een vrouw is op een meer aan het wakeboarden

Simple present vs present continuous

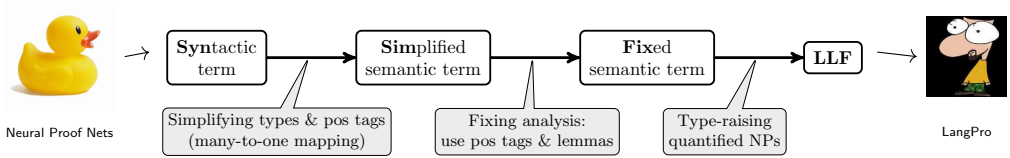
The man is swimming	→	De man is aan het zwemmen
The man is swimming	→	De man zwemt

Stress Testing

	present cont. → present simple			
	Before	After	→	←
BERT	84.55	86.63	93.21	92.43
mBERT	86.11	84.90	94.26	94.52
RobBERT	82.81	81.94	86.16	84.33
	prep. phrase order switch			
BERT	81.03	78.45	85.06	85.06
mBERT	87.93	85.34	85.06	80.46
RobBERT	76.72	75.86	72.41	73.56

Logical Reasoning on Dutch

Return of the Lambdas [Abzianidze and Kogkalidis, 2021] implement a system that uses neurally learnt lambda terms for tackling the SICK-NL dataset:



Example

(nld) mannen zijn hout aan het snijden
(sim) $\mathsf{zijn}^{vp(n(s))}\ (\mathsf{aan\_het}^{vp(vp)}\ (\mathsf{snijden}^{n(vp)}\ \mathsf{hout}^{n}))\ \mathsf{mannen}^{n}$
(fix) $\mathbf{zijn}^{vp(np(s))}\ (\mathbf{aan\_het}^{vp(vp)}\ (\mathbf{snijden}^{np(vp)}\ (\mathbf{een}^{n(np)}\ \mathbf{hout}^{n})))\ (\mathbf{s}^{n(np)}\ \mathbf{man}^{n})$
(llf) $\mathbf{s}^{n(vp(s))}\ \mathbf{man}\ (\mathbf{zijn}\ (\mathbf{aan\_het}\ (\lambda x.\,\mathbf{een}^{n(vp(s))}\ \mathbf{hout}\ (\lambda y.\,\mathbf{snijden}\ y^{np}\ x^{np}))))$

Results

Model	Accuracy	Hybrid
LangPro Σ^2	78.8	–
BERTje	82.0	81.8
RobBERT	81.7	82.6
mBERT	79.9	80.6

Evaluating syntactic patterns

Probing Discontinuity

Goal [Kogkalidis and Wijnholds, 2022] studies BERT embeddings to see the extent to which they contain lexical knowledge about control verbs, and the extent to which they are invariant under word order permutations in the case of verb raising.

The plan

1. Generate test data for verb-subject dependencies in discontinous settings,
2. Generate in a controlled way and generate samples that are naturalistic (i.e. humans can understand them),
3. Training a probing model to recognise verb-subject dependencies,
4. Analyzing the results

Refinement We refine the generation using the ACG format seen on Wednesday, which allows us to carry out analysis in much more detail.

DIY `https://github.com/gijswijnholds/discontinuous-probing`

Generating Test Data: Verb Raising

Crossing Dependencies

'...that John sees Mary teach the kids to cycle'

Generating samples Using an MCFG + annotations indicating verbs and their corresponding subjects:

$$\begin{aligned}
\text{S}(xy_1y_2) &\longleftarrow \text{PREF}(x)\ \text{SUB}(y_1, y_2) && (B_1)\\
\text{SUB}(x, y) &\longleftarrow \text{NP}(x)\ \text{INF}_{iv}(y) && (B_2)\\
\text{SUB}(xy, z) &\longleftarrow \text{NP}(x)\ \text{NP}(y)\ \text{INF}_{tv}(z) && (B_3)\\
\text{SUB}(xz, yu) &\longleftarrow \text{NP}(x)\ \text{RV}(y)\ \text{SUB}(z, u) && (B_4)
\end{aligned}$$

Example

(*a*) de docent ziet [de student] [de hond] [de eend] de oefeningen [helpen] [leren] [eten]

(EN) *the teacher sees [the student] [help] [the dog] [teach] [the duck] [to eat] the exercises*

Generating Test Data: Control Verbs

Understood subject Control verbs pass their own subject/object to the infinitive verbal complement that they select for.

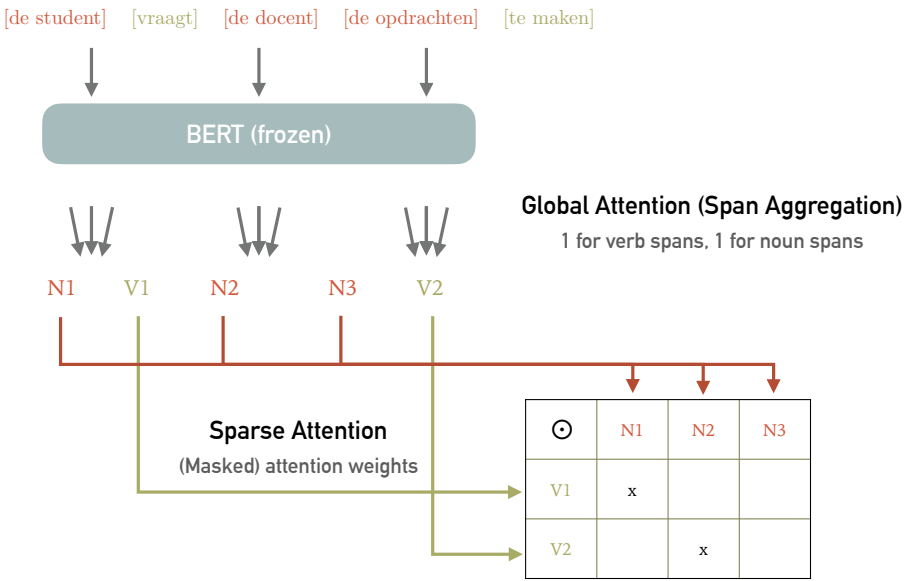
(Simple) Example

(*a*)	[de student]	belooft	de docent	[te vertrekken]
(*b*)	de student	vraagt	[de docent]	[te vertrekken]
(EN)	*the student*	*promises/asks*	*the teacher*	*to leave*

The MCFG

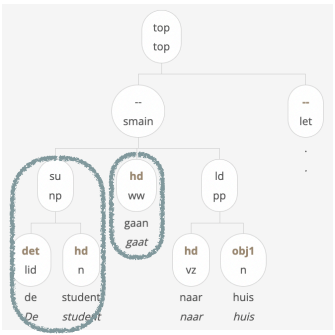
$$
\begin{array}{rcll}
\textsc{s}(xyzu_1u_2) & \longleftarrow & \textsc{np}(x)\ \textsc{tv}(y)\ \textsc{np}(z)\ \textsc{vc}(u_1,u_2) & (A_1)\\
\textsc{s}(xyzuw_1vw_2) & \longleftarrow & \textsc{np}(x)\ \textsc{tv}(y)\ \textsc{np}(z)\ \textsc{np}(u)\ \textsc{cv}(v)\ \textsc{vc}(w_1,w_2) & (A_2)\\
\textsc{vc}(x,y) & \longleftarrow & \textsc{te}(x)\ \textsc{inf}_{iv}(y) & (A_3)\\
\textsc{vc}(zx,y) & \longleftarrow & \textsc{te}(x)\ \textsc{inf}_{tv}(y)\ \textsc{np}(z) & (A_4)\\
\textsc{vc}(xy,zu_0u_1) & \longleftarrow & \textsc{np}(x)\ \textsc{te}(y)\ \textsc{inf}_{c}(z)\ \textsc{vc}(u_0,u_1) & (A_5)\\
\textsc{vc}(xyu,zv_1v_2) & \longleftarrow & \textsc{np}(x)\ \textsc{te}(y)\ \textsc{inf}_{c}(z)\ \textsc{cv}(u)\ \textsc{vc}(v_1,v_2) & (A_6)\\
\hline
\textsc{s}(xyzvu_1u_2) & \longleftarrow & \textsc{np}(x)\ \textsc{tv}(y)\ \textsc{np}(z)\ \textsc{vc}(u_1,u_2)\ \textsc{adv}(v) & (A_1^m)\\
\textsc{s}(vyxzu_1u_2) & \longleftarrow & \textsc{np}(x)\ \textsc{tv}(y)\ \textsc{np}(z)\ \textsc{vc}(u_1,u_2)\ \textsc{adv}(v) & (A_1^i)\\
 & \vdots & &
\end{array}
$$

Probe Design

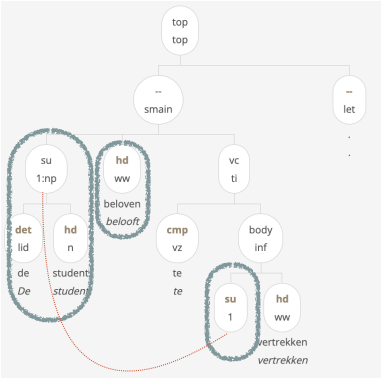


Training Data

Lassy-Small is a gold standard dataset of written Dutch, containing ca. 65k sentences, that include both continuous and discontinuous verb-subject dependencies:



The student goes home



The student promises to leave

Results

Validation vs test results The prober works fine, but our generated test sets are very challenging:

Model	Lassy	Control	Raising
BERTje	97.6	48	43.1
RobBERT	92.5	40.6	29.2

Investigating further Because we are using a grammar that generates syntax trees, we can inspect the results, filtering by the complexity of the generated sentences:

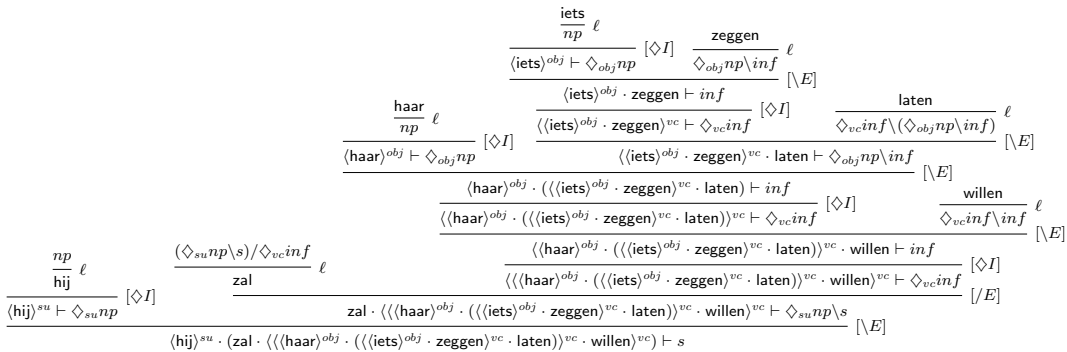
	# Nouns				Tree Depth			Rule					
Model	2	3	4	5	2	3	4	A_1^X	A_2^X	A_3	A_4	A_5	A_6
BERTje	81.1	58.8	50.5	42.9	61.8	52.7	46.8	100	67	43.1	34.6	36.1	27.1
RobBERT	73	52.8	42.4	35.9	58.3	47.2	38.8	93.6	58.1	41.2	19.6	21	17

(a) Control Verb Grammar

	# Nouns				Tree Depth					Rule		
Model	2	3	4	5	2	3	4	5	6	B_2	B_3	B_4
BERTje	75.6	52.4	33.5	25.5	92.2	66.4	40.5	29	23	53.4	53.8	36.7
RobBERT	46.3	37.2	24.5	11.4	65.6	36.9	33.6	19.4	12.6	89.1	24.3	12.9

(b) Verb Raising Grammar

Recap: Doing it the Diamond Way



ACG all the way

$$\lceil\dagger\rceil^{string} = \texttt{hij}\cdot\texttt{zal}\cdot\texttt{haar}\cdot\texttt{iets}\cdot\texttt{willen}\cdot\texttt{laten}\cdot\texttt{zeggen}$$

$$\lceil\dagger\rceil^{sem} = \text{WILL}\ (\text{WANT}\ \Delta^{vc}((\text{LET}\ \Delta^{vc}(\text{SAY}\ \Delta^{obj}\ \text{SOMETHING}))\ \Delta^{obj}\ \text{HER}))\ \Delta^{su}\ \text{HE}$$

$$\lceil\dagger\rceil^{pair} = \text{[(zal,hij),(willen,hij),(laten,hij),(zeggen,haar)]}$$

[Moortgat et al., 2022]

Doing it the Diamond Way: Samples

273 Samples

AST	$x_3\ (x_0\ d_7)\ (h_1\ (f_1\ d_5\ d_1))$
Surface	`hij zal dreigen iets te zeggen`
Semantics	(`zal` ($\Diamond_{vc}$(`dreigen` ($\Diamond_{vc}$(`te` ($\Diamond_{vc}$(`zeggen` ($\Diamond_{obj1}$`iets`)))))))) ($\Diamond_{su}$`hij`)
Pairing	`[(zal,hij),(dreigen,hij),(zeggen,hij)]`

AST	$r_0\ (g_0\ (h_0(h_1\ (f_1\ d_5\ d_1))\ d_6)\ d_2)$
Surface	`hij zal iets willen proberen te zeggen`
Semantics	(`zal` ($\Diamond_{vc}$(`willen` ($\Diamond_{vc}$(`proberen` ($\Diamond_{vc}$(`te` ($\Diamond_{vc}$(`zeggen` ($\Diamond_{obj1}$`iets`)))))))))) ($\Diamond_{su}$`hij`)
Pairing	`[(zal,hij),(willen,hij),(proberen,hij),(zeggen,hij)]`

AST	$x_3\ (x_1\ (x_0\ d_7)\ d_2)\ (h_1\ (f_1\ d_5\ d_1))$
Surface	`hij zal willen dreigen iets te zeggen`
Semantics	(`zal` ($\Diamond_{vc}$(`willen` ($\Diamond_{vc}$(`dreigen` ($\Diamond_{vc}$(`te` ($\Diamond_{vc}$`zeggen` ($\Diamond_{obj1}$`iets`)))))))))) ($\Diamond_{su}$`hij`)
Pairing	`[(zal,hij),(willen,hij),(dreigen,hij),(zeggen,hij)]`

$\vdots$

Doing it the Diamond Way: Populating the lexicon

The lexicon

Category	Description	Examples
INF0	intransitive infinitive	vertrekken, stemmen, verliezen, ...
INF1	transitive infinitive with inanimate object	zeggen, begrijpen, merken, ...
INF1A	transitive infinitive, animate object	ontmoeten, bedanken, kennen, ...
IVR0	obligatory verb raiser	willen, zullen, moeten, ...
IVR1	obligatory verb raiser, subject flipper	laten, doen
IVR2	non-obligatory verb raiser	proberen, weigeren, trachten, ...
INF2	extraposition	proberen, weigeren, trachten, ...
INF3	extraposition, object control	verzoeken, dwingen, verplichten, ...
INF4	extraposition, subject control	beloven, verzekeren, zweren, ...
OBJ1A	animate direct object	Karin, Wouter, ...
OBJ1I	inanimate direct object	iets, veel, een ding, ...
OBJ2	indirect object	Karin, Wouter, ...

Results

Validation vs test results The probe does not perform as well on the generated data:

	Validation set (Lassy)	Test set (generated)
Accuracy	97.60	79.47
Random Baseline	13.24	39.24

By number of nouns Again, the more nouns, the more challenging the test case:

Number of nouns	2	3	4
Accuracy	86.87	75.66	68.76
Random Baseline	50.00	33.33	25.00

Raising, Extraposition, and Infinitives

Raising

		$\texttt{OBJ}_2$	$\texttt{OBJ}_1^I$	$\texttt{IVR}_0$	$\texttt{IVR}_1$	$\texttt{INF}_1$
hij	zal	haar	iets	willen	laten	zeggen
he	*will*	*her*	*something*	*want*	*let*	*say*

Extraposition

		$\texttt{OBJ}_2$	$\texttt{INF}_3$	$\texttt{OBJ}_1^I$	TE	$\texttt{INF}_1$
hij	zal	haar	dwingen	iets	te	zeggen
he	*will*	*her*	*force*	*something*	*to*	*say*

By verbal type Extraposition slightly easier, as there is no cluster. Infinitives are worse, because they typically are far removed from their understood subjects:

Verbal type	Raising	Extraposition	Infinitive
Accuracy	81.00	87.03	68.77
Random	39.86	38.27	39.24

Verb Dominance

Dominance

		$\texttt{OBJ}_2$	$\texttt{OBJ}_1^I$	$\texttt{IVR}_0$	$\texttt{IVR}_1$	$\texttt{INF}_1$
hij	zal	haar	iets	willen	laten	zeggen
he	*will*	*her*	*something*	*want*	*let*	*say*

		$\texttt{INF}_2$	$\texttt{OBJ}_2$	$\texttt{OBJ}_1^I$	TE	$\texttt{IVR}_1$	$\texttt{INF}_1$
hij	zal	proberen	haar	iets	te	laten	zeggen
he	*will*	*try*	*her*	*something*	*to*	*let*	*say*

Results Verbs under the scope of an extraposition verb are more challenging!

Dominated verb, grouped by verbal type

Dominated by raising	**Overall**	**Raising**	**Extraposition**	**Infinitive**
Accuracy	76.18	76.23	77.76	74.68
Random Baseline	39.86	41.23	38.60	39.76
Dominated by extraposition				
Accuracy	66.70	67.35	85.35	59.62
Random Baseline	38.27	38.60	36.84	38.30

Verb Dominance

An orthogonal view we distinguish the different subcategories of verbs that govern other verbs:

Dominating verb, by subcategory

Dominated by raising	**Overall**	**IVR0**	**IVR1**	**IVR2**
Accuracy	76.18	78.54	71.41	77.95
Random Baseline	39.86	41.06	37.09	41.05
Dominated by extraposition	**Overall**	**INF2**	**INF3**	**INF4**
Accuracy	66.70	86.74	57.12	47.12
Random Baseline	38.27	42.58	35.13	35.13

It's all about control

		$\texttt{OBJ}_2$	$\texttt{INF}_4/\texttt{INF}_3$	$\texttt{OBJ}_1^I$	TE	$\texttt{INF}_1$
hij	zal	haar	beloven/dwingen	iets	te	zeggen
he	*will*	*her*	*promise/force*	*something*	*to*	*say*

Verb Dominance

An orthogonal view we distinguish the different subcategories of verbs that govern other verbs:

Dominating verb, by subcategory

Dominated by raising	**Overall**	**IVR0**	**IVR1**	**IVR2**
Accuracy	76.18	78.54	71.41	77.95
Random Baseline	39.86	41.06	37.09	41.05
Dominated by extraposition	**Overall**	**INF2**	**INF3**	**INF4**
Accuracy	66.70	86.74	57.12	47.12
Random Baseline	38.27	42.58	35.13	35.13

It's all about control

		$\texttt{OBJ}_2$	$\texttt{INF}_{3/4}$	$\texttt{OBJ}_1^I$	TE	$\texttt{INF}_1$
hij	zal	haar	[MASK]	iets	te	zeggen
he	*will*	*her*	[MASK]	*something*	*to*	*say*

	Object (INF3)	**Subject (INF4)**	**Other**
Control preference of governed verb	35.08	59.20	5.72

Word order variations

Semantic equivalence We can group samples that originate from different ASTs, have identical semantics, but different surface realizations:

a. *hij zal haar proberen*[IVR2] *te willen ontmoeten*
he will her try to want meet

b. *hij zal proberen*[INF2] *haar te willen ontmoeten*
he will try her to want meet
'he will try to want to meet her'

Results Extra confirmation that extraposition is the easier construction to recognize.

	Context in the sentence		
Raising construction	**Above**	**Verb**	**Below**
Accuracy	95.09	86.22	78.15
Random Baseline	42.54	41.47	41.44
Extraposition construction			
Accuracy	96.49	93.04	78.50
Random Baseline	42.54	41.48	41.44

Epilogue

Conclusion of the day

- Evaluating tensor-based models is possible, but quickly gets intractable,
- We seem to always 'lose' to BERT,
- But BERT is not the answer: we showed that BERT embeddings are not (inherently) capable of recognizing verb-subject dependencies (in Dutch).
- Types to the rescue? We will have to put effort in downstream tasks that can make good use of lambda terms, or require subtle linguistic reasoning

Conclusion of the week

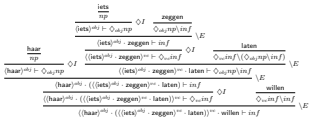
- Compositionality as a guiding principle for semantic reasoning
- Once the toolkit is in place, we can address the linguistic subtleties that models like BERT may not understand all too well
- The dream: integrating compositional tools/methods with neural reasoning
- For example:
  1. Using syntactic/semantic terms as input on downstream tasks
  2. Merging syntactic/semantic terms with BERT embeddings [Tziafas et al., 2021]

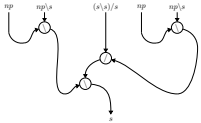
DIY What would you do?

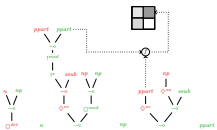
- See next week's workshop for the latest developments in the field
- Or read up at: http://dx.doi.org/10.4204/EPTCS.366

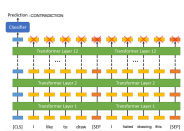
We thank the NWO for supporting this project: https://compositioncalculus.sites.uu.nl/

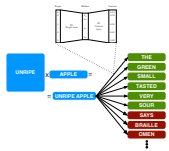
Apotheosis



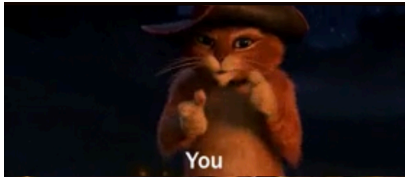








But the greatest ingredient:



(and funding)

References

Lasha Abzianidze and Konstantinos Kogkalidis. A logic-based framework for natural language inference in dutch. Computational Linguistics in the Netherlands Journal, 11:35–58, 2021.

Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. A large annotated corpus for learning natural language inference. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 632–642, 2015.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. Distributional semantics in technicolor. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 136–145, 2012.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. e-snli: Natural language inference with natural language explanations. Advances in Neural Information Processing Systems, 31, 2018.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. Xnli: Evaluating cross-lingual sentence representations. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2475–2485, 2018.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context: The concept revisited. In Proceedings of the 10th international conference on World Wide Web, pages 406–414, 2001.

Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. Simverb-3500: A large-scale evaluation set of verb similarity. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 2173–2182, 2016.

Edward Grefenstette and Mehrnoosh Sadrzadeh. Experimental support for a categorical compositional distributional model of meaning. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. Proceedings of the 2011 Conference on Empirical Methods in Natural Language . . . , 2011.

Felix Hill, Roi Reichart, and Anna Korhonen. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. Computational Linguistics, 41(4):665–695, 2015.

Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. Prior disambiguation of word tensors for constructing sentence vectors. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 1590–1601, 2013.

Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Stephen Pulman. Separating disambiguation from composition in distributional semantics. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning, pages 114–123, 2013.

Konstantinos Kogkalidis and Gijs Wijnholds. Discontinuous constituency and bert: A case study of dutch. In Findings of the Association for Computational Linguistics: ACL 2022, pages 3776–3785, 2022.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. A sick cure for the evaluation of compositional distributional semantic models.

In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pages 216–223, 2014.

Tom McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3428–3448, 2019.

Dmitrijs Milajevs, Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Matthew Purver. Evaluating neural word representations in tensor-based compositional settings. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 708–719, 2014.

George A Miller and Walter G Charles. Contextual correlates of semantic similarity. Language and cognitive processes, 6(1):1–28, 1991.

Jeff Mitchell and Mirella Lapata. Vector-based models of semantic composition. In proceedings of ACL-08: HLT, pages 236–244, 2008.

Jeff Mitchell and Mirella Lapata. Composition in distributional models of semantics. Cognitive science, 34(8):1388–1429, 2010.

Michael Moortgat, Konstantinos Kogkalidis, and Gijs Wijnholds. Untitled. Forthcoming, 2022.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. Stress test evaluation for natural language inference. In Proceedings of the 27th International Conference on Computational Linguistics, pages 2340–2353, 2018.

Kyle Richardson, Hai Hu, Lawrence Moss, and Ashish Sabharwal. Probing natural language inference models through semantic fragments. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, pages 8713–8721, 2020.

Herbert Rubenstein and John B Goodenough. Contextual correlates of synonymy. Communications of the ACM, 8(10):627–633, 1965.

Aarne Talman and Stergios Chatzikyriakidis. Testing the generalization power of neural network models across nli benchmarks. In Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pages 85–94, 2019.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. What do you learn from context? probing for sentence structure in contextualized word representations. arXiv preprint arXiv:1905.06316, 2019.

Giorgos Tziafas, Konstantinos Kogkalidis, Gijs Wijnholds, and Michael Moortgat. Improving bert pretraining with syntactic supervision. arXiv preprint arXiv:2104.10516, 2021.

Gijs Wijnholds and Michael Moortgat. Sick-nl: A dataset for dutch natural language inference. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 1474–1479, 2021.

Gijs Wijnholds and Mehrnoosh Sadrzadeh. Evaluating composition models for verb phrase elliptical sentence embeddings. In NAACL HLT 2019-2019 Conference of the North American

Chapter of the Association for Computational Linguistics: Human Language Technologies-Proceedings of the Conference, volume 2019, pages 261–271. Association for Computational Linguistics (ACL), 2019.

Gijs Wijnholds, Mehrnoosh Sadrzadeh, and Stephen Clark. Representation learning for type-driven composition. In Proceedings of the 24th Conference on Computational Natural Language Learning, pages 313–324, 2020.

Gijs Jasper Wijnholds. A Compositional Vector Space Model of Ellipsis and Anaphora. PhD thesis, Queen Mary University of London, 2020.

Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1112–1122, 2018.

Hitomi Yanaka, Koji Mineshima, Daisuke Bekki, Kentaro Inui, Satoshi Sekine, Lasha Abzianidze, and Johan Bos. Can neural networks understand monotonicity reasoning? In Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pages 31–40, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4804. URL https://aclanthology.org/W19-4804.

Dongqiang Yang and David MW Powers. Verb similarity on the taxonomy of WordNet. Masaryk University, 2006.