Weighted posets

Learning surface order from dependency trees

William Dyer

Oracle Corp

18th International Workshop on Treebanks and Linguistic Theories,
Syntax Fest, 2019

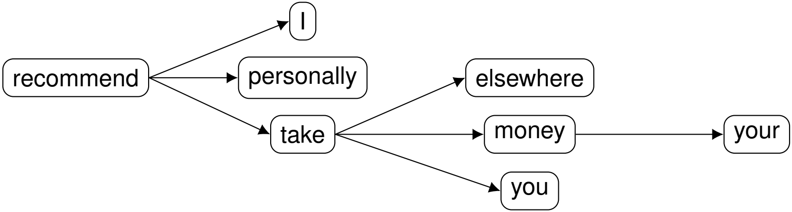
Outline

1. Task
   - Syntactic tree to surface realization
   - Previous work
2. Methodology
   - Weighted posets (sorted)
   - Syntactic embeddings
   - Graph neural network
   - Example
3. Results
4. Discussion

Outline

From syntactic tree to surface realization

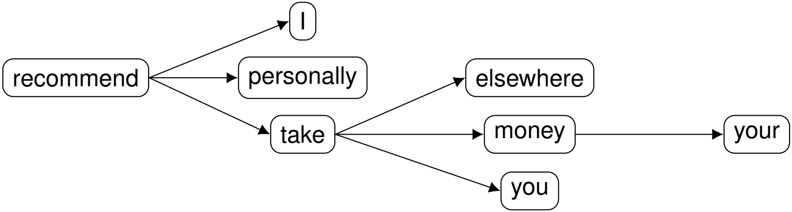
(a) syntactic tree (DAG)



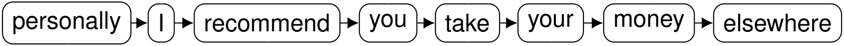
(b) surface realization

From syntactic tree to surface realization

(a) syntactic tree (DAG)

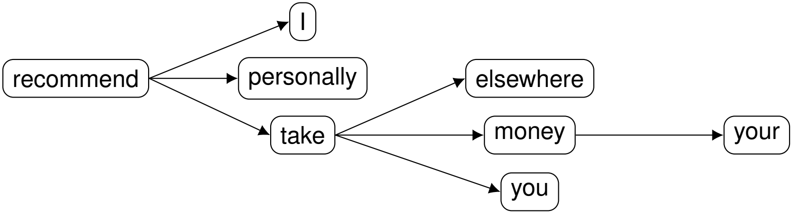


(b) surface realization

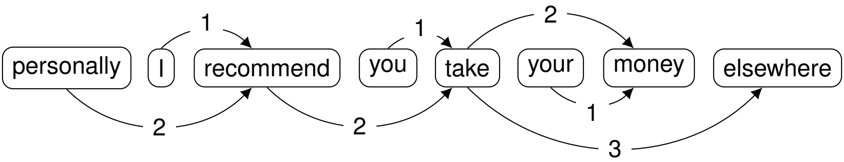


From syntactic tree to surface realization

(a) syntactic tree (DAG)



(b′) surface realization (poset)



Outline

1. Task
   - Syntactic tree to surface realization
   - Previous work
2. Methodology
   - Weighted posets (sorted)
   - Syntactic embeddings
   - Graph neural network
   - Example
3. Results
4. Discussion

Previous linguistic work

- Specific constituents
  - demonstratives, numerals, adjectives (Greenberg, 1963)
  - manner, place, time (Boisson, 1981)
  - adjective order restrictions (Scott, 2002)
  - complements and adjuncts
- General tree principles
  - "what belongs together semantically is also placed close together" (Behaghel, 1932)
  - projectivity (Marcus, 1965)
  - Head Proximity (Rijkhoff, 1986)
  - Early Immediate Constituents (Hawkins, 1994)
  - Dependency Distance Minimization (Hudson, 1995)
  - Dependency Locality Theory (Gibson, 2000)
  - Minimize Domains (Hawkins, 2004)
  - Uniform Information Density (Jaeger and R. Levy, 2006)

Previous linguistic work

- Specific constituents
  - demonstratives, numerals, adjectives (Greenberg, 1963)
  - manner, place, time (Boisson, 1981)
  - adjective order restrictions (Scott, 2002)
  - complements and adjuncts
- General tree principles
  - “what belongs together semantically is also placed close together” (Behaghel, 1932)
  - projectivity (Marcus, 1965)
  - Head Proximity (Rijkhoff, 1986)
  - Early Immediate Constituents (Hawkins, 1994)
  - Dependency Distance Minimization (Hudson, 1995)
  - Dependency Locality Theory (Gibson, 2000)
  - Minimize Domains (Hawkins, 2004)
  - Uniform Information Density (Jaeger and R. Levy, 2006)

Previous linguistic work

- Sequential order
  - "old concepts come before new ones" (Behaghel, 1932)
  - "most important information first" (cf. Gundel, 1988)
  - precedence relations (Gerdes and Kahane, 2001; Kahane and Lareau, 2016)
  - extend DDm with info-theoretic measures (Dyer, 2018; Hahn et al., 2018)

Previous NLG work

- Bag of words
  - "for language is not merely a bag of words but a tool with particular properties which have been fashioned in the course of its use" (Harris, 1954)
- SR '18: First Multilingual Surface Realisation Shared Task (Mille et al., 2018)
  - determine word order and inflections
  - bigram language model with binary neural-net classification (Puzikov and Gurevych, 2018)
  - seq-to-seq MT model augmented with synthetic/outside data (Elder and Hokamp, 2018)
  - sort dependents into preceding or following groups, then by syntactic category or using max entropy classifier (Castro Ferreira et al., 2018)
  - incrementally linearize words based on dependency structure and distance (King and White, 2018)

Previous NLG work

- Bag of words
  - “for language is not merely a bag of words but a tool with particular properties which have been fashioned in the course of its use” (Harris, 1954)
- SR ‘18: First Multilingual Surface Realisation Shared Task (Mille et al., 2018)
  - determine word order and inflections
  - bigram language model with binary neural-net classification (Puzikov and Gurevych, 2018)
  - seq-to-seq MT model augmented with synthetic/outside data (Elder and Hokamp, 2018)
  - sort dependents into preceding or following groups, then by syntactic category or using max entropy classifier (Castro Ferreira et al., 2018)
  - incrementally linearize words based on dependency structure and distance (King and White, 2018)

Previous NLG work

- Bag of words
  - "for language is not merely a bag of words but a tool with particular properties which have been fashioned in the course of its use" (Harris, 1954)
- SR '18: First Multilingual Surface Realisation Shared Task (Mille et al., 2018)
  - determine word order and inflections
  - bigram language model with binary neural-net classification (Puzikov and Gurevych, 2018)
  - seq-to-seq MT model augmented with synthetic/outside data (Elder and Hokamp, 2018)
  - sort dependents into preceding or following groups, then by syntactic category or using max entropy classifier (Castro Ferreira et al., 2018)
  - incrementally linearize words based on dependency structure and distance (King and White, 2018)

Previous NLG work

- Bag of words
  - "for language is not merely a bag of words but a tool with particular properties which have been fashioned in the course of its use" (Harris, 1954)
- SR '18: First Multilingual Surface Realisation Shared Task (Mille et al., 2018)
  - determine word order and inflections
  - bigram language model with binary neural-net classification (Puzikov and Gurevych, 2018)
  - seq-to-seq MT model augmented with synthetic/outside data (Elder and Hokamp, 2018)
  - sort dependents into preceding or following groups, then by syntactic category or using max entropy classifier (Castro Ferreira et al., 2018)
  - incrementally linearize words based on dependency structure and distance (King and White, 2018)

Previous NLG work

- Bag of words
  - “for language is not merely a bag of words but a tool with particular properties which have been fashioned in the course of its use” (Harris, 1954)
- SR ‘18: First Multilingual Surface Realisation Shared Task (Mille et al., 2018)
  - determine word order and inflections
  - bigram language model with binary neural-net classification (Puzikov and Gurevych, 2018)
  - seq-to-seq MT model augmented with synthetic/outside data (Elder and Hokamp, 2018)
  - sort dependents into preceding or following groups, then by syntactic category or using max entropy classifier (Castro Ferreira et al., 2018)
  - incrementally linearize words based on dependency structure and distance (King and White, 2018)

Previous NLG work

- Bag of words
  - “for language is not merely a bag of words but a tool with particular properties which have been fashioned in the course of its use” (Harris, 1954)
- SR ‘18: First Multilingual Surface Realisation Shared Task (Mille et al., 2018)
  - determine word order and inflections
  - bigram language model with binary neural-net classification (Puzikov and Gurevych, 2018)
  - seq-to-seq MT model augmented with synthetic/outside data (Elder and Hokamp, 2018)
  - sort dependents into preceding or following groups, then by syntactic category or using max entropy classifier (Castro Ferreira et al., 2018)
  - incrementally linearize words based on dependency structure and distance (King and White, 2018)

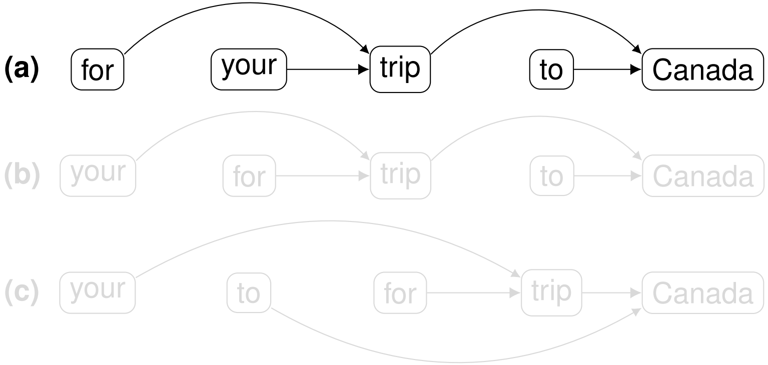
Outline

Poset

- partially ordered set (poset)
  - *for* $\prec$ *trip*, *your* $\prec$ *trip*, *trip* $\prec$ *Canada*, *to* $\prec$ *Canada*

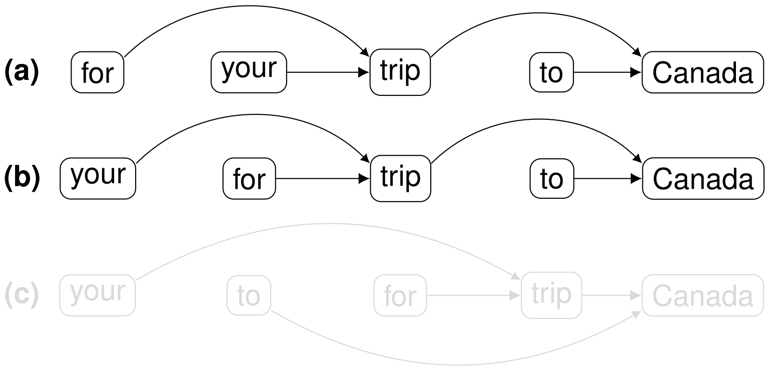
Poset

- partially ordered set (poset)
  - *for* $\prec$ *trip*, *your* $\prec$ *trip*, *trip* $\prec$ *Canada*, *to* $\prec$ *Canada*



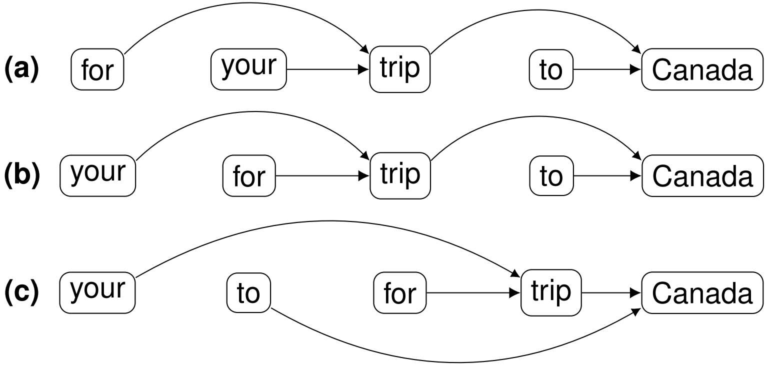
Poset

- partially ordered set (poset)
  - *for* $\prec$ *trip*, *your* $\prec$ *trip*, *trip* $\prec$ *Canada*, *to* $\prec$ *Canada*



Poset

- partially ordered set (poset)
  - *for* $\prec$ *trip*, *your* $\prec$ *trip*, *trip* $\prec$ *Canada*, *to* $\prec$ *Canada*

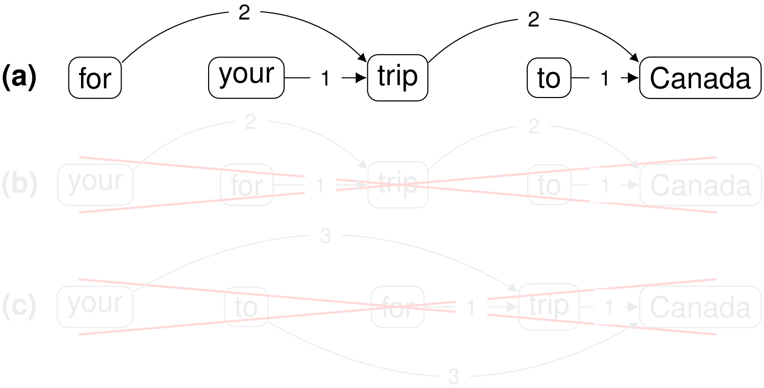


Weighted poset

- edge-weighted poset
  - *for* $\overset{2}{\prec}$ *trip*, *your* $\overset{1}{\prec}$ *trip*, *trip* $\overset{2}{\prec}$ *Canada*, *to* $\overset{1}{\prec}$ *Canada*

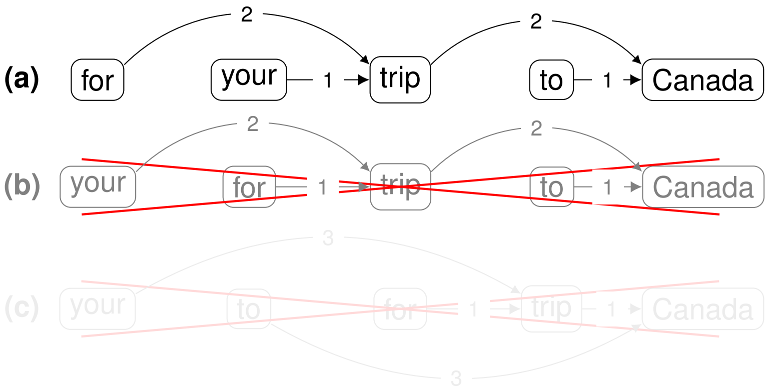
Weighted poset

- edge-weighted poset
  - *for* $\overset{2}{\prec}$ *trip*, *your* $\overset{1}{\prec}$ *trip*, *trip* $\overset{2}{\prec}$ *Canada*, *to* $\overset{1}{\prec}$ *Canada*



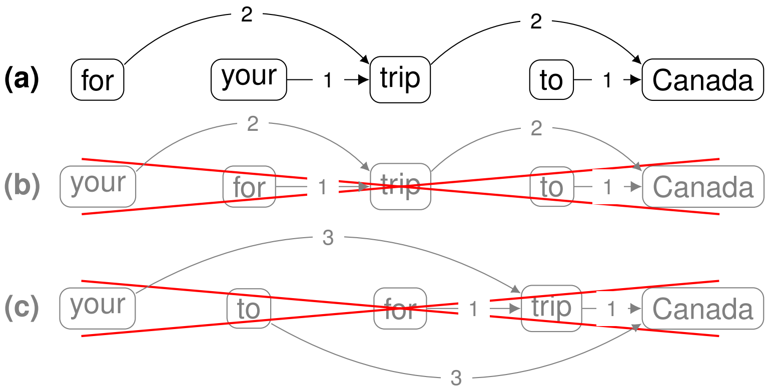
Weighted poset

- edge-weighted poset
  - *for* $\overset{2}{\prec}$ *trip*, *your* $\overset{1}{\prec}$ *trip*, *trip* $\overset{2}{\prec}$ *Canada*, *to* $\overset{1}{\prec}$ *Canada*



Weighted poset

- edge-weighted poset
  - *for* $\overset{2}{\prec}$ *trip*, *your* $\overset{1}{\prec}$ *trip*, *trip* $\overset{2}{\prec}$ *Canada*, *to* $\overset{1}{\prec}$ *Canada*



Topologically sorting a weighted poset

Algorithm 1: Given an edge-weighted *poset*, construct a total order such that nodes with smallest weights are adjacent.

```
function WEIGHTED_TOPO_SORT(poset)
    order ← ∅                                          ▷ empty directed graph to hold totally ordered set
    for (u, v, w_uv) ∈ poset do
        w_sum ← 0                                      ▷ a sum of traversed weights
        if u ∈ order then
            while w_uv > w_sum do                      ▷ traverse successors of u
                s ← order.u.successor
                w_us ← order[u][s].weight
                w_sum ← w_sum + w_us
                if w_uv < w_sum then
                    u ← s                              ▷ u becomes its successor s
            w_vs ← w_sum − w_uv                        ▷ w_vs is how much w_sum overshot w_uv
            order.UPDATE_EDGE(u, s, _) ←               ▷ change existing (u, s)...
                [(u, v, w_us − w_vs), (v, s, w_vs)]    ▷ ... to (u, v) and (v, s) and update weights
        else if v ∈ order then
            while w_uv > w_sum do                      ▷ traverse predecessors of v
                p ← order.v.predecessor
                w_pv ← order[p][v].weight
                w_sum ← w_sum + w_pv
                if w_uv < w_sum then
                    v ← p                              ▷ v becomes its predecessor p
            w_pu ← w_sum − w_uv                        ▷ w_pu is how much w_sum overshot w_uv
            order.UPDATE_EDGE(p, v, _) ←               ▷ change existing (p, v)...
                [(p, u, w_pu), (u, v, w_pv − w_pu)]    ▷ ... to (p, u) and (u, v) and update weights
        else
            order.ADD_EDGE(u, v, w_uv)
    return TOPO_SORT(order)                            ▷ return topological sort of order graph
```

Topologically sorting a weighted poset

Algorithm 1: Given an edge-weighted *poset*, construct a total order such that nodes with smallest weights are adjacent.

```
function WEIGHTED_TOPO_SORT(poset)
    order ← ∅                                        ▷ empty directed graph to hold totally ordered set
    for (u, v, w_uv) ∈ poset do
        w_sum ← 0                                    ▷ a sum of traversed weights
        if u ∈ order then
            while w_uv > w_sum do                    ▷ traverse successors of u
                s ← order.u.successor
                w_us ← order[u][s].weight
                w_sum ← w_sum + w_us
                if w_uv < w_sum then
                    u ← s                            ▷ u becomes its successor s
            w_vs ← w_sum − w_uv                      ▷ w_vs is how much w_sum overshot w_uv
            order.UPDATE_EDGE(u, s, _) ←             ▷ change existing (u, s)...
                [(u, v, w_us − w_vs), (v, s, w_vs)]  ▷ ... to (u, v) and (v, s) and update weights
        else if v ∈ order then
            while w_uv > w_sum do                    ▷ traverse predecessors of v
                p ← order.v.predecessor
                w_pv ← order[p][v].weight
                w_sum ← w_sum + w_pv
                if w_uv < w_sum then
                    v ← p                            ▷ v becomes its predecessor p
            w_pu ← w_sum − w_uv                      ▷ w_pu is how much w_sum overshot w_uv
            order.UPDATE_EDGE(p, v, _) ←             ▷ change existing (p, v)...
                [(p, u, w_pu), (u, v, w_pv − w_pu)]  ▷ ... to (p, u) and (u, v) and update weights
        else
            order.ADD_EDGE(u, v, w_uv)
    return TOPO_SORT(order)                          ▷ return topological sort of order graph
```

Outline

Syntactic embeddings

- Distributional hypothesis
  - “you shall know a word by the company it keeps” (Firth, 1957)
- Represent words as dense vectors (via NN)
  - `dancing [0.43 1.91 -0.22 0.95 -0.89 ...]`
  - similar words have cosine-similar vectors
- Context
  - linear (continuous bag-of-words) – `word2vec` (Mikolov et al., 2013)
  - `dancing` similar to `singing`, `dance`, `dances`, `dancers`
  - syntactic – `word2vecf` (O. Levy and Goldberg, 2014)
  - `dancing` similar to `singing`, `rapping`, `miming`, `busking`

Syntactic embeddings

- Distributional hypothesis
  - "you shall know a word by the company it keeps" (Firth, 1957)
- Represent words as dense vectors (via NN)
  - `dancing [0.43 1.91 -0.22 0.95 -0.89 ...]`
  - similar words have cosine-similar vectors
- Context
  - linear (continuous bag-of-words) – `word2vec` (Mikolov et al., 2013)
  - `dancing` similar to `singing`, `dance`, `dances`, `dancers`
  - syntactic – `word2vecf` (O. Levy and Goldberg, 2014)
  - `dancing` similar to `singing`, `rapping`, `miming`, `busking`

Syntactic embeddings

- Distributional hypothesis
  - “you shall know a word by the company it keeps” (Firth, 1957)
- Represent words as dense vectors (via NN)
  - `dancing [0.43 1.91 -0.22 0.95 -0.89 ...]`
  - similar words have cosine-similar vectors
- Context
  - linear (continuous bag-of-words) – `word2vec` (Mikolov et al., 2013)
  - `dancing` similar to `singing`, `dance`, `dances`, `dancers`
  - syntactic – `word2vecf` (O. Levy and Goldberg, 2014)
  - `dancing` similar to `singing`, `rapping`, `miming`, `busking`

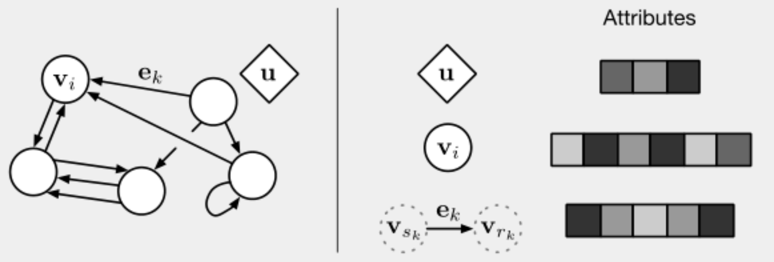
Syntactic embeddings

- Distributional hypothesis
  - "you shall know a word by the company it keeps" (Firth, 1957)
- Represent words as dense vectors (via NN)
  - `dancing [0.43 1.91 -0.22 0.95 -0.89 ...]`
  - similar words have cosine-similar vectors
- Context
  - linear (continuous bag-of-words) – `word2vec` (Mikolov et al., 2013)
  - `dancing` similar to `singing`, `dance`, `dances`, `dancers`
  - syntactic – `word2vecf` (O. Levy and Goldberg, 2014)
  - `dancing` similar to `singing`, `rapping`, `miming`, `busking`

Outline

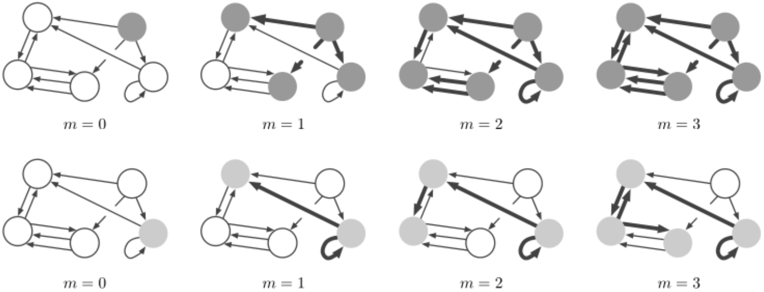
Graph neural network (GNN)

- Graph Nets (GN) framework (Battaglia et al., 2018)



Graph neural network (GNN)

- Graph Nets (GN) framework (Battaglia et al., 2018)
- Message-passing neural network (MPNN) (Gilmer et al., 2017)
- Spatial-based graph convolutions and pooling (Wu et al., 2019)



Outline

Example - `word2vecf`

(a) *[input] conllu file (abridged)*

9	for	for	ADP	11	case
10	your	you	PRON	11	nmod:poss
11	trip	trip	NOUN	3	obl
12	to	to	ADP	13	case
13	Canada	Canada	PROPN	11	nmod

(b) *[output] syntactic embeddings*

for|ADP|case	[1.69 -0.51 ...]
your|PRON|nmod:poss	[0.92 -0.61 ...]
trip|NOUN|obl	[0.17 -0.11 ...]
to|ADP|case	[1.24 -0.59 ...]
canada|PROPN|nmod	[0.05 -0.05 ...]
ADP|case	[0.12 -0.80 ...]
ADP	[0.10 -0.07 ...]

Example - word2vecf

(a) *[input] conllu file (abridged)*

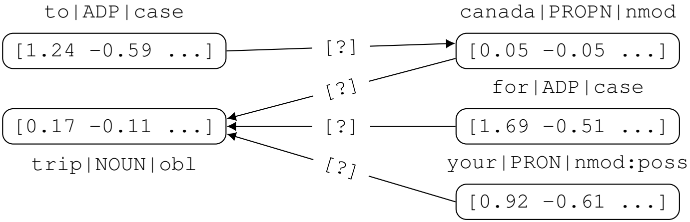
9	for	for	ADP	11	case
10	your	you	PRON	11	nmod:poss
11	trip	trip	NOUN	3	obl
12	to	to	ADP	13	case
13	Canada	Canada	PROPN	11	nmod

(b) *[output] syntactic embeddings*

```
for|ADP|case              [1.69  -0.51  ...]
your|PRON|nmod:poss       [0.92  -0.61  ...]
trip|NOUN|obl             [0.17  -0.11  ...]
to|ADP|case               [1.24  -0.59  ...]
canada|PROPN|nmod         [0.05  -0.05  ...]
ADP|case                  [0.12  -0.80  ...]
ADP                       [0.10  -0.07  ...]
```

Example - GNN

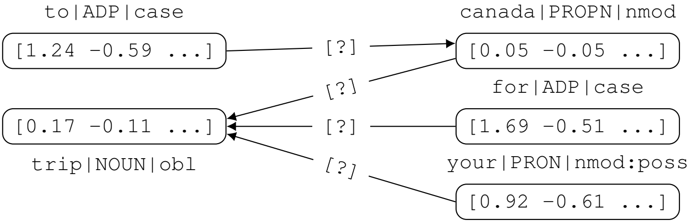
(c) *[input] directed* `networkx` *graph of dependency tree*



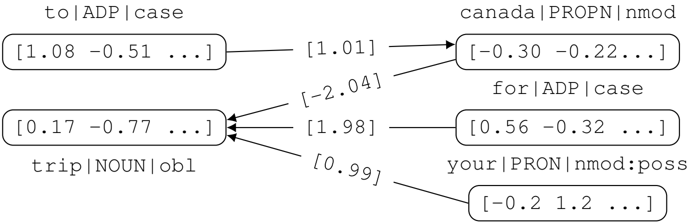
(d) *[output] directed graph with learned edge attributes*

Example - GNN

(c) *[input] directed* `networkx` *graph of dependency tree*

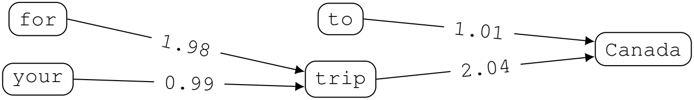


(d) *[output] directed graph with learned edge attributes*



Example - topological sort

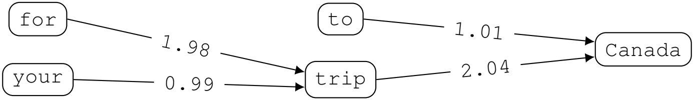
(e) *[input] edge-weighted poset*



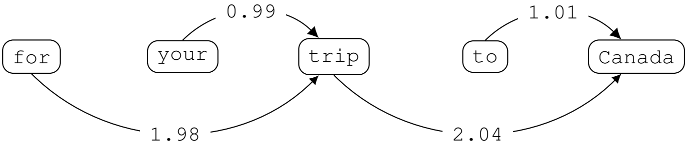
(f) *[output] topological sort*

Example - topological sort

(e) *[input] edge-weighted poset*



(f) *[output] topological sort*



Baseline

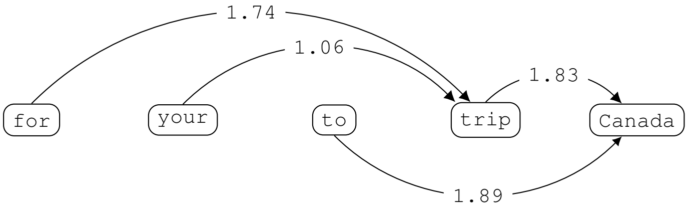
- Average dependency distance
  - for|ADP|case $\overset{1.74}{\prec}$ trip|NOUN|obl
  - your|PRON|nmod:poss $\overset{1.06}{\prec}$ trip|NOUN|obl
  - trip|NOUN|obl $\overset{1.83}{\prec}$ PROPN|nmod
  - to|ADP|case $\overset{1.89}{\prec}$ PROPN|nmod

Baseline

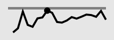
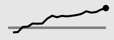
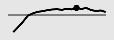
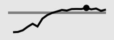
- Average dependency distance
  - `for|ADP|case` $\overset{1.74}{\prec}$ `trip|NOUN|obl`
  - `your|PRON|nmod:poss` $\overset{1.06}{\prec}$ `trip|NOUN|obl`
  - `trip|NOUN|obl` $\overset{1.83}{\prec}$ `PROPN|nmod`
  - `to|ADP|case` $\overset{1.89}{\prec}$ `PROPN|nmod`

Baseline

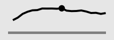
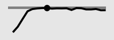
- Average dependency distance
  - for|ADP|case $\overset{1.74}{\prec}$ trip|NOUN|obl
  - your|PRON|nmod:poss $\overset{1.06}{\prec}$ trip|NOUN|obl
  - trip|NOUN|obl $\overset{1.83}{\prec}$ PROPN|nmod
  - to|ADP|case $\overset{1.89}{\prec}$ PROPN|nmod



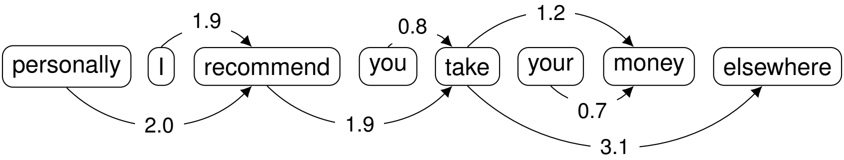
Results

	SPEARMAN'S ρ [-1,1]			PROJECTIVITY [0,1]			
	AVG		GNN	AVG		GNN	UD
Afrikaans	0.707		**0.773**	0.530		**0.650**	0.939
Armenian	0.628		**0.672**	0.413		**0.585**	0.987
Czech	0.665		0.659	0.359		**0.469**	0.982
English	0.634		**0.775**	0.496		**0.680**	0.995
French	0.677		**0.729**	0.531		**0.669**	0.998
Greek	0.731		**0.754**	0.503		**0.651**	0.996
Hungarian	0.635		0.609	0.440		**0.598**	0.969

Results

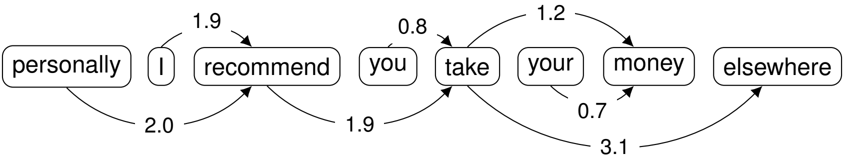
	SPEARMAN'S ρ [-1,1]			PROJECTIVITY [0,1]			
	AVG		GNN	AVG		GNN	UD
Irish	0.674		**0.753**	0.461		**0.603**	0.978
Italian	0.657		**0.796**	0.482		**0.651**	0.996
Latin	0.614		0.582	0.613		**0.729**	0.855
Maltese	0.729		**0.750**	0.498		**0.682**	0.995
Slovenian	0.549		**0.567**	0.663		**0.798**	0.967
Telugu	0.916		**0.931**	0.925		**0.971**	0.997
Uyghur	0.728		0.727	0.629		**0.762**	0.976

Discussion



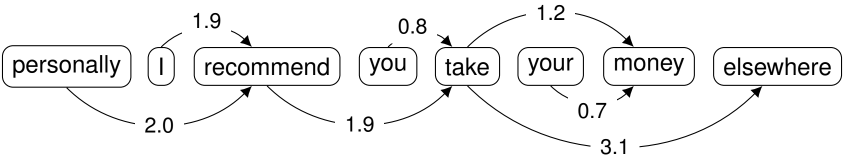
- Engineering
  - not E2E, but using ML to address parts of problem
  - useful data structure for representing surface realizations
  - entirely within dependency framework
- What is GNN learning?
  - relative individual dependency-distance **tolerances** ...
  - based on **context** of words (embeddings) and structure (MPNN)
- Emergent projectivity rate
  - no baked-in notion or representation of projectivity
  - rate reflects (approaches) that of training data

Discussion



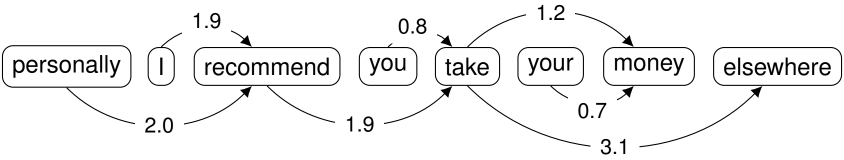
- Engineering
  - not E2E, but using ML to address parts of problem
  - useful data structure for representing surface realizations
  - entirely within dependency framework
- What is GNN learning?
  - relative individual dependency-distance **tolerances** ...
  - based on **context** of words (embeddings) and structure (MPNN)
- Emergent projectivity rate
  - no baked-in notion or representation of projectivity
  - rate reflects (approaches) that of training data

Discussion



- Engineering
  - not E2E, but using ML to address parts of problem
  - useful data structure for representing surface realizations
  - entirely within dependency framework
- What is GNN learning?
  - relative individual dependency-distance **tolerances** ...
  - based on **context** of words (embeddings) and structure (MPNN)
- Emergent projectivity rate
  - no baked-in notion or representation of projectivity
  - rate reflects (approaches) that of training data

Discussion



- Engineering
  - not E2E, but using ML to address parts of problem
  - useful data structure for representing surface realizations
  - entirely within dependency framework
- What is GNN learning?
  - relative individual dependency-distance **tolerances** ...
  - based on **context** of words (embeddings) and structure (MPNN)
- Emergent projectivity rate
  - no baked-in notion or representation of projectivity
  - rate reflects (approaches) that of training data

Future study

- improve design of GNN
- customize hyperparameters based on corpus
- use newer embedding frameworks
- develop/find efficient algorithm for sorting weighted posets
- apply weighted posets to study graph-theoretic measures

Works cited I

Peter W. Battaglia et al. (2018). “Relational inductive biases, deep learning, and graph networks”. In: *arXiv:1806.01261 [cs, stat]*.

Otto Behaghel (1932). *Deutsche Syntax eine geschichtliche Darstellung*. Heidelberg: Carl Winters Unversitätsbuchhandlung.

Claude Boisson (1981). “Hiérarchie universelle des spécifications de temps, de lieu, et de manière.”. In: *Confluents* 7, pp. 69–124.

Thiago Castro Ferreira, Sander Wubben, and Emiel Krahmer (2018). “Surface Realization Shared Task 2018 (SR18): The Tilburg University Approach”. In: *Proceedings of the First Workshop on Multilingual Surface Realisation*. Melbourne, Australia: ACL, pp. 35–8.

William Dyer (2018). “Integration complexity and the order of cosisters”. In: *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*. Brussels, Belgium: ACL, pp. 55–65.

Henry Elder and Chris Hokamp (2018). “Generating High-Quality Surface Realizations Using Data Augmentation and Factored Sequence Models”. In: *Proceedings of the First Workshop on Multilingual Surface Realisation*. Melbourne, Australia: ACL, pp. 49–53.

Ramon Ferrer-i-Cancho (2017). “Towards a theory of word order. Comment on" Dependency distance: a new perspective on syntactic patterns in natural language" by Haitao Liu et al”. In: *Physics of Life Reviews*.

John Rupert Firth (1957). “A synopsis of linguistic theory 1930-1955”. In: *Studies in Linguistic Analysis*. Oxford: Philological Society, pp. 1–32.

Kim Gerdes and Sylvain Kahane (2001). “Word Order in German: A Formal Dependency Grammar Using a Topological Hierarchy”. In: *Proceedings of 39th Annual Meeting of the ACL*. Toulouse, France: ACL, pp. 220–7.

Edward Gibson (2000). “The dependency locality theory: A distance-based theory of linguistic complexity”. In: *Image, language, brain*, pp. 95–126.

Works cited II

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl (2017). "Neural Message Passing for Quantum Chemistry". In: *arXiv:1704.01212 [cs]*.

Joseph Greenberg (1963). "Some universals of grammar with particular reference to the order of meaningful elements". In: *Universals of Grammar*. Ed. by Joseph Greenberg. Cambridge, Massachusetts: MIT Press, pp. 73–113.

Jeanette Gundel (1988). "Universals of topic-comment stucture". In: *Studies in Syntactic Typology*. Ed. by Michael Hammond, Edith Moravcsik, and Jessica Wirth. Philadelphia: John Benjamins Publishing, pp. 209–39.

Michael Hahn, Judith Degen, Noah Goodman, Dan Jurafsky, and Richard Futrell (2018). "An Information-Theoretic Explanation of Adjective Ordering Preferences". In: *Proceedings of the 40th annual conference of the Cognitive Science Society*. London: Cognitive Science Society.

Zellig S. Harris (1954). "Distributional Structure". In: *WORD* 10.2, pp. 146–62.

John A. Hawkins (1994). *A Performance Theory of Order and Constituency*. Cambridge: Cambridge University Press.

John A. Hawkins (2004). *Efficiency and Complexity in Grammars*. Oxford: Oxford University Press.

Richard Hudson (1995). "Measuring syntactic difficulty". URL: http://dickhudson.com/wp-content/uploads/2013/07/Difficulty.pdf.

T. Florian Jaeger and Roger Levy (2006). "Speakers optimize information density through syntactic reduction". In: *Advances in neural information processing systems*, pp. 849–56.

Sylvain Kahane and François Lareau (2016). "Word Ordering as a Graph Rewriting Process". In: *Formal Grammar*. Ed. by Annie Foret, Glyn Morrill, Reinhard Muskens, Rainer Osswald, and Sylvain Pogodalla. Springer Berlin Heidelberg, pp. 216–39.

Works cited III

David King and Michael White (2018). “The OSU Realizer for SRST ‘18: Neural Sequence-to-Sequence Inflection and Incremental Locality-Based Linearization”. In: *Proceedings of the First Workshop on Multilingual Surface Realisation*. Melbourne, Australia: ACL, pp. 39–48.

Omer Levy and Yoav Goldberg (2014). “Dependency-Based Word Embeddings.”. In: *ACL (2)*. Citeseer, pp. 302–8.

Solomon Marcus (1965). “Sur la notion de projectivité”. In: *Mathematical Logic Quarterly* 11.2, pp. 181–92.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean (2013). “Efficient estimation of word representations in vector space”. In: *arXiv:1301.3781 [cs]*.

Simon Mille, Anja Belz, Bernd Bohnet, Yvette Graham, Emily Pitler, and Leo Wanner (2018). “The First Multilingual Surface Realisation Shared Task (SR’18): Overview and Evaluation Results”. In: *Multilingual Surface Realisation: Shared Task and Beyond: Proceedings of the Workshop*. Multilingual Surface Realisation: Shared Task and Beyond. Melbourne, Australia: ACL, pp. 1–12.

Yevgeniy Puzikov and Iryna Gurevych (2018). “BinLin: A Simple Method of Dependency Tree Linearization”. In: *Proceedings of the First Workshop on Multilingual Surface Realisation*. Melbourne, Australia: ACL, pp. 13–28.

Jan Rijkhoff (1986). “Word Order Universals Revisited: The Principle of Head Proximity”. In: *Belgian Journal of Linguistics* 1, pp. 95–125.

Gary-John Scott (2002). “Stacked adjectival modification and the structure of nominal phrases”. In: *Functional Structure in DP and IP: The Cartography of Syntactic Structures*. Vol. 1. New York: Oxford University Press, pp. 91–120.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu (2019). “A Comprehensive Survey on Graph Neural Networks”. In: *arXiv:1901.00596 [cs, stat]*.

Thank you!

Weighted posets
Learning surface order from dependency trees

William Dyer

william.dyer@oracle.com
researchgate.net/profile/William_Dyer5
linkedin.com/in/william-dyer/