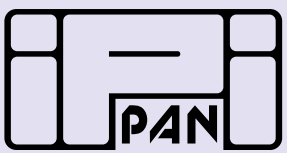
Informacje lingwistyczne w wektorowych reprezentacjach zdań

Alina Wróblewska
Katarzyna Krasnowska-Kieraś



INSTITUTE OF COMPUTER SCIENCE
POLISH ACADEMY OF SCIENCES
ul. Jana Kazimierza 5, 01-248 Warszawa

Warszawa, 27 stycznia 2020

Spis treści

1 Wprowadzenie

2 Metodologia badawcza

3 Eksperymenty

4 Wyniki

5 Podsumowanie

Wprowadzenie

- Projekt **Scwad**: Kompozycyjno-dystrybucyjne modelowanie semantyki języka polskiego
- Kompozycyjno-dystrybucyjny model semantyczny:
  - „komponuje” (łączy) dystrybucyjne wektory słów (zanurzenia słowne, ang. word embeddings),
  - generuje reprezentację wektorową znaczenia zdania (zanurzenie zdaniowe, ang. sentence embedding) lub frazy.

Motywacja

- Modelowanie języka naturalnego sieciami neuronowymi:
  - ✓ obniżenie kosztów inżynierii cech,
  - ? sieć to tzw. black box – nie wiemy, które informacje wejściowe są wykorzystywane w trenowaniu modelu.
- W większości zadań NLP modele neuronowe działają bardzo dobrze, a najczęściej lepiej niż dotychczasowe modele.
- Badania nad „zawartością" zanurzeń (tzn. cechami segmentów lub zdań zachowanych w zanurzeniach) są niezwykle interesujące i pożądane.

Założenia badawcze

- **Pytanie badawcze**: czy informacje lingwistyczne są zakodowane w zanurzeniach zdaniowych (ang. sentence embeddings)?
- **Metody badawcze**: metody oparte na uniwersalnych zadaniach próbkowania (ang. probing tasks) i na zaawansowanych zadaniach NLP (ang. downstream tasks).
- **Zakres badań**: eksperymenty z różnymi typami zanurzeń zdaniowych dla języka angielskiego i polskiego.

Wyniki badań zostały opublikowane w materiałach konferencji ACL 2019 w artykule pt. „Empirical Linguistic Study of Sentence Embeddings".

Motywacja językowa

- Eksperymenty zostały przeprowadzone na dwóch typologiczne różnych językach:
  - angielskim – język izolacyjny,
  - polskim – język fleksyjny.
- Weryfikacja metod NLP na dwóch różniących się językach jest bardziej obiektywna.
- W tych dwóch językach istnieją porównywalne korpusy powiązania semantycznego i wynikania tekstowego.

Uniwersalność badań

- Zaproponowane testy próbkowania są uniwersalne:
  - opierają się na schemacie Universal Dependencies [Nivre et al., 2016],
  - zdania do testów próbkowania są wybierane na podstawie ich automatycznych rozbiorów UD,
  - analogiczne dane mogą być generowane dla każdego języka z bankiem drzew UD.

Oryginalny wkład

1. Opracowanie uniwersalnych testów próbkowania do analizowania zawartości zanurzeń zdaniowych.
2. Upublicznienie zbiorów danych do testów próbkowania dla angielskiego i polskiego:
   `http://git.nlp.ipipan.waw.pl/Scwad/SCWAD-probing-data`.
3. Przeprowadzenie serii eksperymentów empirycznych i szczegółowa analiza wyników.

Spis treści

1. Wprowadzenie
2. Metodologia badawcza
3. Eksperymenty
4. Wyniki
5. Podsumowanie

Metoda testów próbkowania (1)

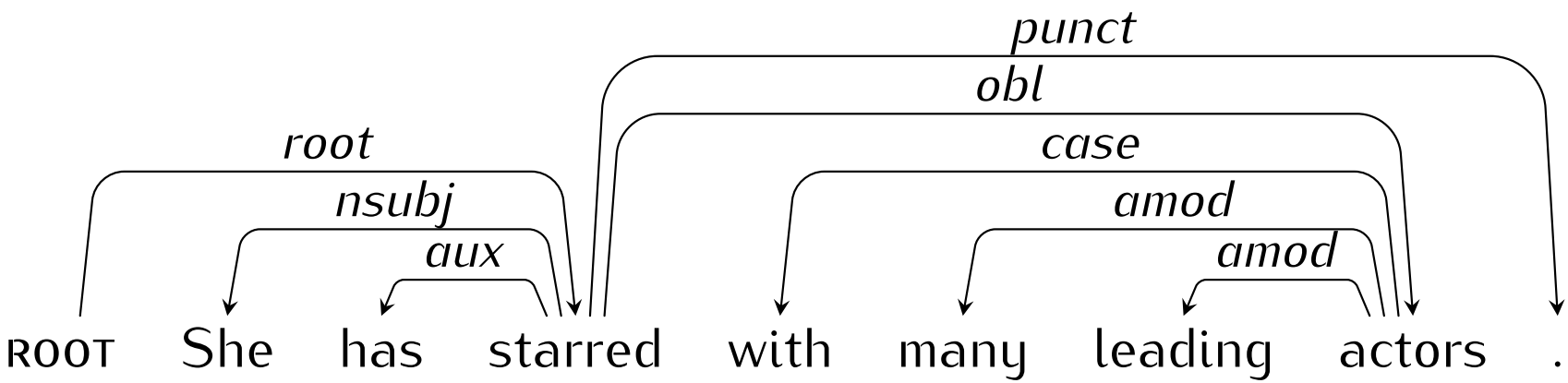
- Zadanie próbkowania to problem klasyfikacji, który koncentruje się na prostych właściwościach lingwistycznych zdania [Conneau et al., 2018].
- Dane do testów próbkowania:
  - zawierają pary: zdanie + kategoria (etykieta cechy lingwistycznej),
  - są automatycznie ekstrahowane z korpusu automatycznie sparsowanych zdań,
  - procedura ekstrakcji opiera się na regułach kompatybilnych ze schematem Universal Dependencies [Nivre et al., 2016].

Metoda testów próbkowania (2)

- Klasyfikator:
  - jest trenowany i testowany na zanurzeniach zdaniowych (ang. sentence embeddings) reprezentujących zdania,
  - jeśli cecha lingwistyczna jest zakodowana w zanurzeniu zdania i klasyfikator nauczy się, w jaki sposób została zakodowana, to dokona poprawnej klasyfikacji.
- Jakość klasyfikatora (accuracy) jest pośrednio wyznacznikiem tego, czy cecha lingwistyczna jest zakodowana w zanurzeniach zdaniowych.

Testy cech powierzchniowych (1)

1. **SentLen** (długość zdania) – klasyfikacja zdań na podstawie ich długości:
   - 6 klas długości: **0**: (3, 5), **1**: (6, 8), **2**: (9, 11), **3**: (12, 14), **4**: (15, 17), **5**: (18, 20), **6**: (21, 23).
   - Przykład: zdanie ma kategorię **1**, bo zawiera 8 segmentów.

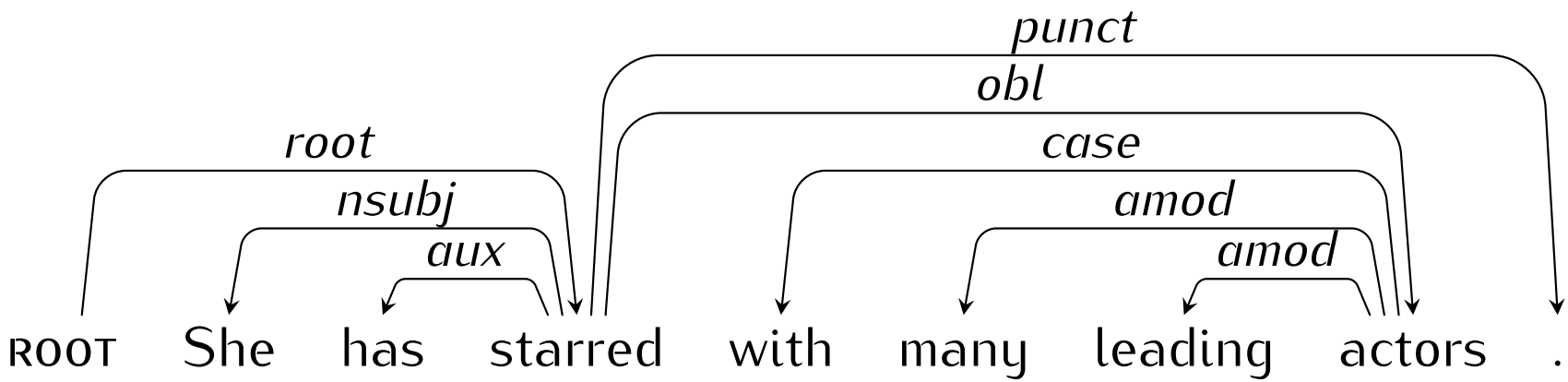


Testy cech powierzchniowych (2)

2. **WC** (ang. word content) – klasyfikacja zdań zawierających jedno ze wstępnie wybranych 750 słów:
   - słowa zostały wybrane z listy frekwencyjnej słów w korpusie – pozycje od 2001 do 2750,
   - kategorie odpowiadają wybranym słowom.

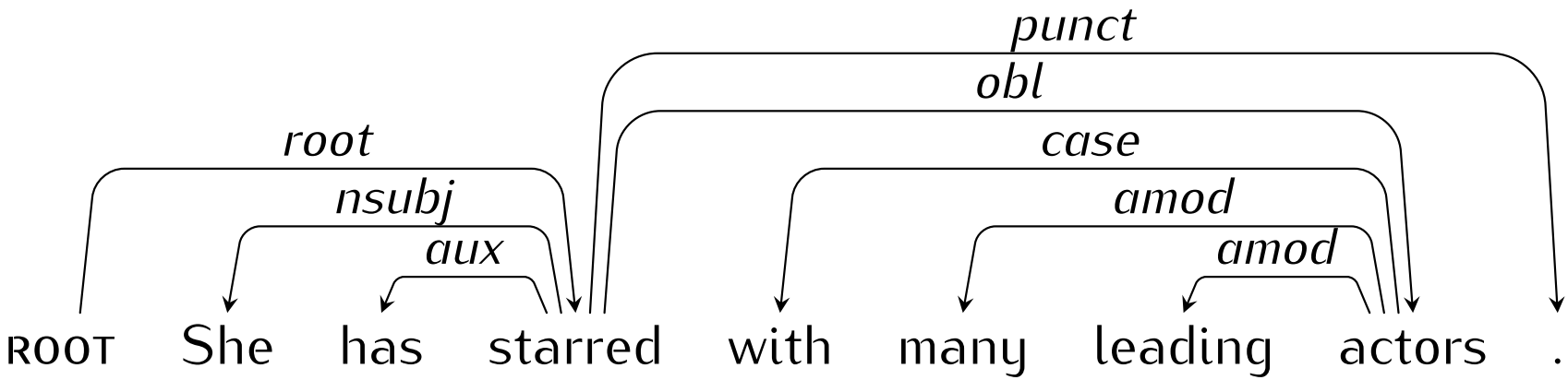
Testy cech kompozycyjnych (1)

3. **TreeDepth** (wysokość drzewa zależnościowego) – klasyfikacja zdań zgodnie z wysokością ich drzew zależnościowych:
   - Dane są dekorelowane w odniesieniu do długości zdania.
   - Przykład: drzewo ma wysokość 3, ponieważ ścieżka od korzenia do każdego innego wierzchołka łączy co najwyżej 3 wierzchołki.



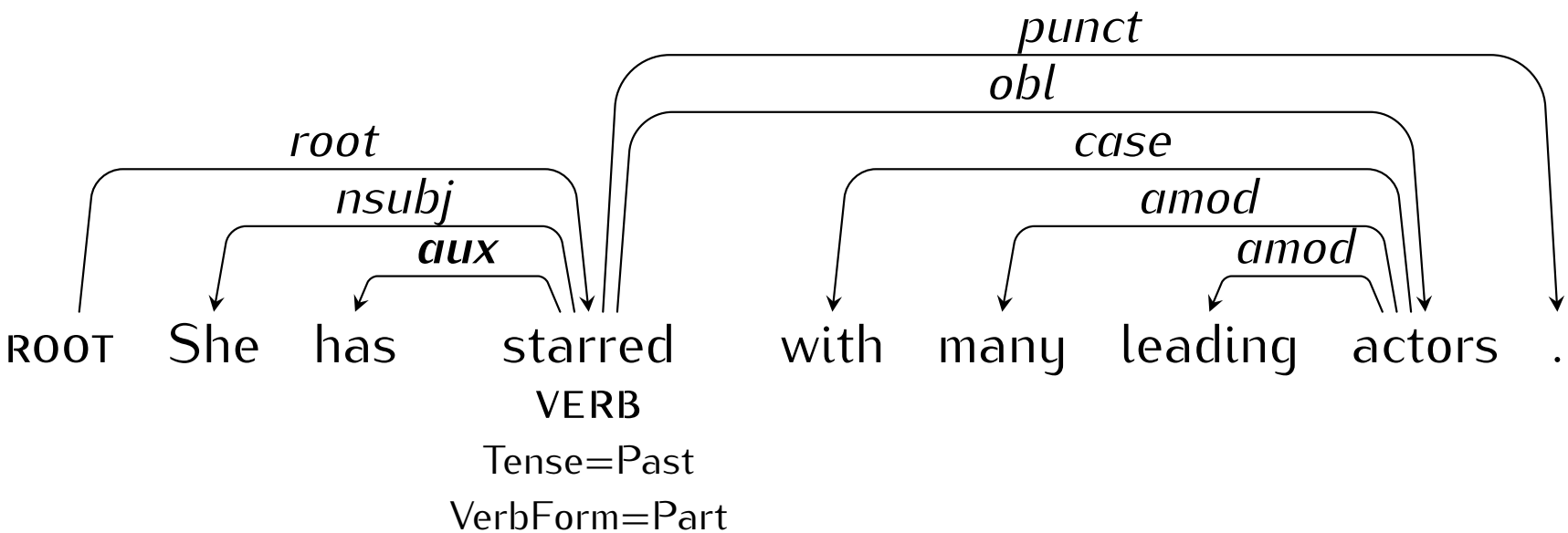
Testy cech kompozycyjnych (2)

4. **TopDeps** (schemat zależnościowy orzeczenia głównego) – klasyfikacja zdań zgodnie z multizbiorem etykiet podrzędników orzeczenia (tj. podrzędnika ROOT-a).
   - Kolejność podrzędników nie jest brana pod uwagę.
   - Znaki interpunkcyjne z etykietą punct są odrzucane.
   - Multizbiory odpowiadają mniej więcej strukturom predykatywno-argumentowym.
   - 20 klas dla każdego języka: 19 najczęstszych schematów zależnościowych i klasa {OTHER}.
   - Przykład: zdanie ma kategorię {aux nsubj obl}.



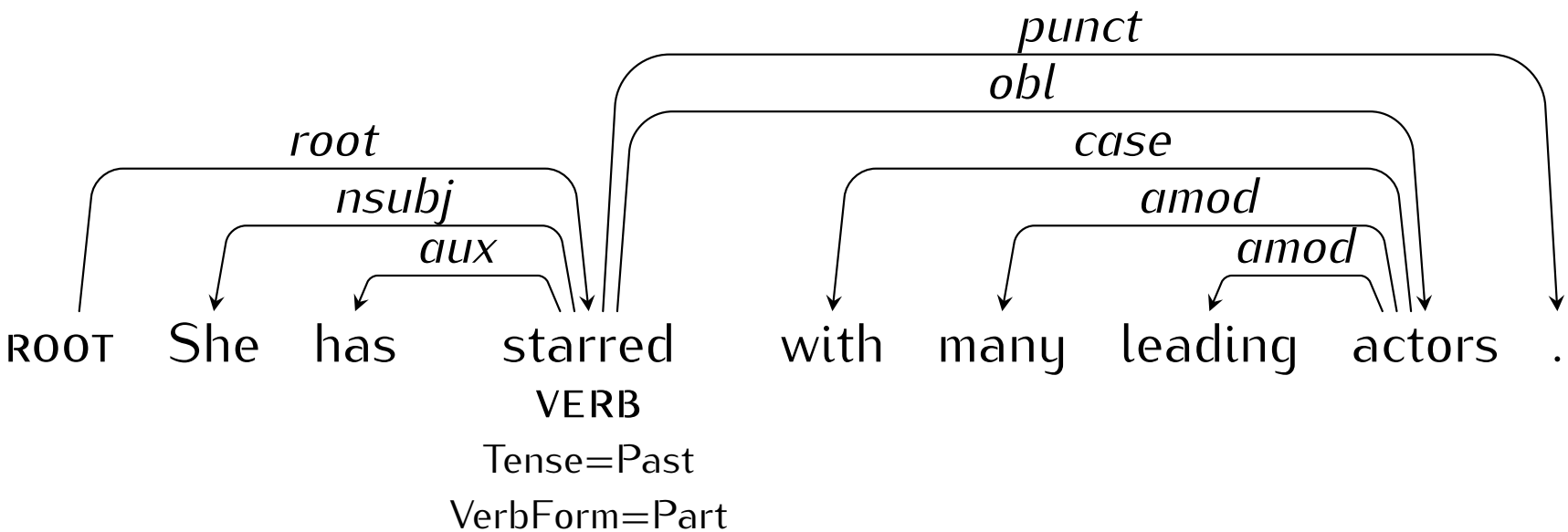
Testy cech kompozycyjnych (3)

5. **Passive** (mowa niezależna) – klasyfikacja zdań zgodnie z kategorią strony gramatycznej orzeczenia głównego.
   - Klasy: **0** (non-passive) i **1** (passive).
   - Przykład: zdanie ma kategorię **0**, bo czasownik posiłkowy nie ma etykiety aux:pass, a czasownik główny nie ma cechy Voice=Pass.



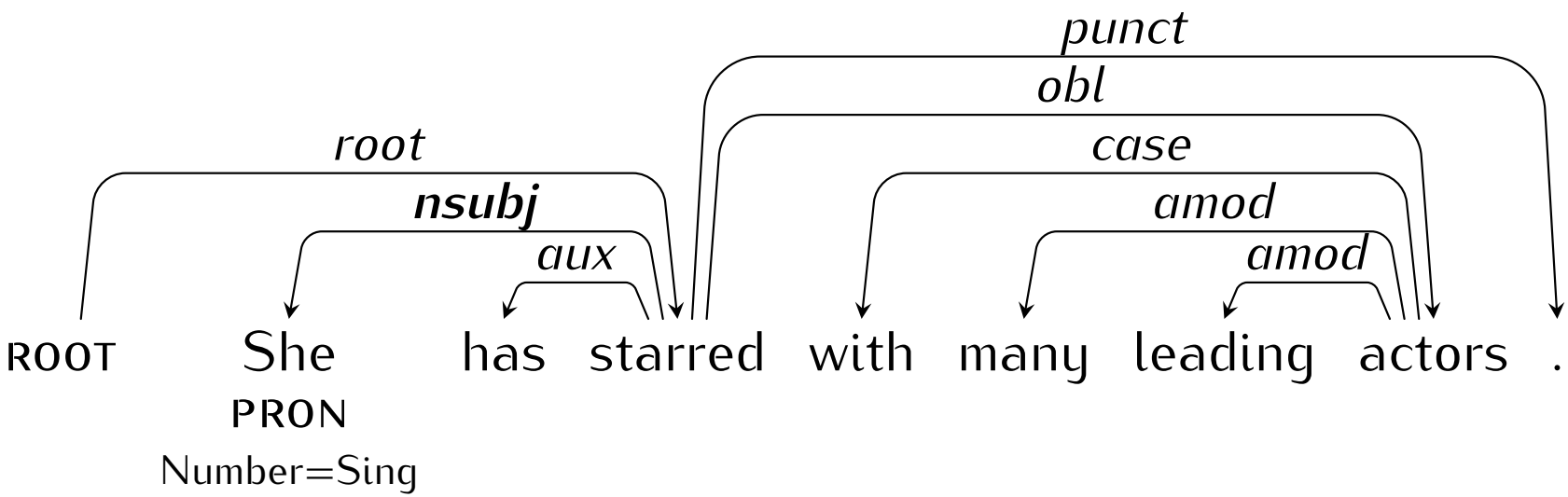
Testy cech kompozycyjnych (4)

6. **Tense** (czas gramatyczny) – klasyfikacja zdań zgodnie z kategorią czasu gramatycznego orzeczenia głównego.
   - Klasy: **pres** (czas teraźniejszy) i **past** (czas przeszły).
   - Przykład: zdanie ma kategorię **past**, bo orzeczenie ma uniwersalną część mowy VERB i cechę Tense=Past.



Testy cech kompozycyjnych (5)

7. **SubjNum** (liczba gramatyczna podmiotu) – klasyfikacja zdań zgodnie z kategorią liczby gramatycznej podmiotów rzeczownikowych (nsubj) orzeczeń głównych.
   - Klasy: **sing** (liczba pojedyncza) i **plur** (liczba mnoga).
   - Przykład: zdanie ma kategorię **sing**, bo zaimek *She* ma cechę Number=Sing.

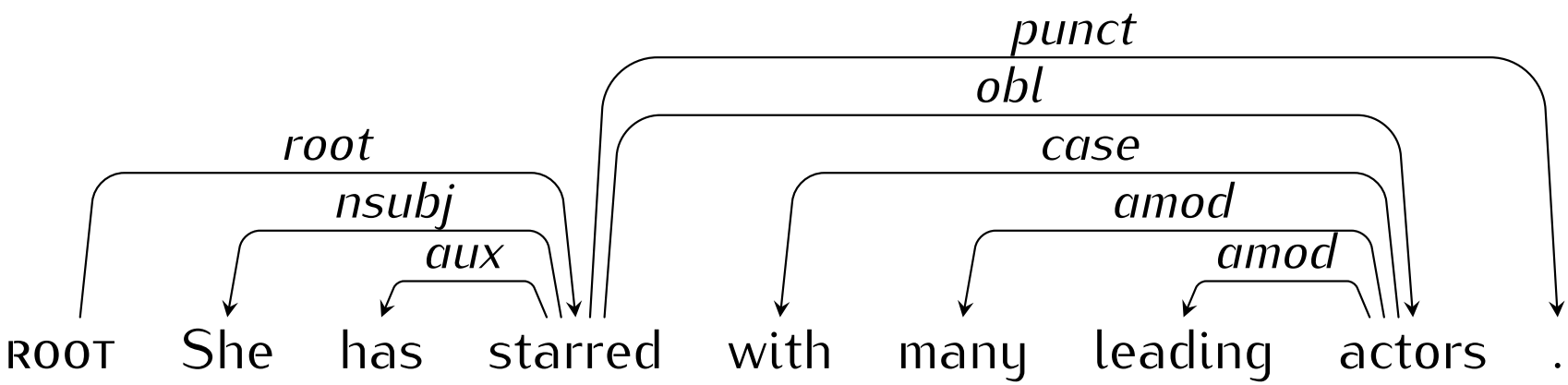


Testy cech kompozycyjnych (6)

8. **ObjNum** (liczba gramatyczna dopełnienia) – klasyfikacja binarna zdań zgodnie z kategorią liczby gramatycznej rzeczownikowego dopełnienia bliższego (obj) orzeczeń głównych.
    - Klasy: **sing** (liczba pojedyncza) i **plur** (liczba mnoga).

Testy cech kompozycyjnych (7)

9. **SentType** (typ zdania) – klasyfikacja zdań zgodnie z ich typem.
   - Typy zdań:
     - **inter** dla zdań pytajnych (np. *Do you like him?*),
     - **imper** dla zdań rozkazujących (e.g. *Get out of here!*),
     - **other** dla zdań oznajmujących (e.g. *He likes her.*) i zdań wykrzyknikowych (e.g. *What a liar!*).
   - Przykład: zdanie ma kategorię **other**.



Metoda testu użycia (ang. downstream task)

- **Powiązanie semantyczne** (ang. semantic relatedness)
  - szacowanie stopnia jakiegokolwiek powiązania leksykalnego lub funkcyjnego między dwoma segmentami, frazami, lub zdaniami,
  - miary ewaluacyjne: współczynniki korelacji r Pearsona i ρ Spearmana.
- **Wynikanie tekstowe** (ang. textual entailment)
  - szacowanie, czy znaczenie jednego zdania wynika ze znaczenia innego zdania,
  - klasy wynikania: *entailment*, *contradiction*, and *neutral*,
  - miara ewaluacyjna: *accuracy*.

Spis treści

1. Wprowadzenie
2. Metodologia badawcza
3. Eksperymenty
4. Wyniki
5. Podsumowanie

Uniwersalne dane do testów próbkowania

- Zbiory danych do testów próbkowania pochodzą z polsko-angielskiego korpusu Paralela [Pęzik, 2016].
- Wszystkie zdania zostały przetworzone na drzewa UD:
  - UDPipe[1] [Straka i Straková, 2017]: segmentacja.
  - COMBO[2] [Rybak i Wróblewska, 2018]: tagowanie, analiza morfologiczna i parsowanie zależnościowe.
- Dane do testów próbkowania są ekstrahowane z drzew UD za pomocą reguł kompatybilnych ze schematem anotacyjnym Universal Dependencies [Nivre et al., 2016].
- Zbiory danych są zbalansowane pod względem kategorii.
- Każdy zbiór danych składa się z 75k przykładów uczących i po 7,5k przykładów walidacyjnych i testowych.

[1] `https://github.com/ufal/udpipe/releases/tag/v1.2.0`
[2] `https://github.com/360er0/COMBO`

Dane do testów użycia

- **SICK**[3] [Bentivogli et al., 2014]:
  - 10k par zdań angielskich,
  - każda para ma przypisany stopień powiązania semantycznego i etykietę wynikania.
- **CDSCorpus**[4] [Wróblewska i Krasnowska-Kieraś, 2017]:
  - 10k par zdań polskich,
  - każda para ma przypisany stopień powiązania semantycznego i dwukierunkową etykietę wynikania.

[3] `http://clic.cimec.unitn.it/composes/materials/SICK.zip`

[4] `http://git.nlp.ipipan.waw.pl/Scwad/SCWAD-CDSCorpus`

SentEval toolkit

- **SentEval** [Conneau i Kiela, 2018] umożliwia testowanie reprezentacji wektorowych zdań w scenariuszach typu „probing” i „downstream”:
  - generuje wektory dla zdań,
  - trenuje klasyfikator z zanurzeniami zdaniowymi na wejściu i etykietami na wyjściu,
  - przeprowadza ewaluację.
- Dane w SentEval-u:
  - nowe dane do testów próbkowania,
  - dodany korpus CDS,
  - zachowany korpus SICK.

Zanurzenia zdaniowe

- Testujemy trzy typy zanurzeń zdaniowych:
  - zanurzenia uzyskane przez uśrednione albo maksymalne łącznie elementów wstępnie wytrenowanych zanurzeń słów lub kontekstowych zanurzeń słów.
  - zanurzenia oszacowane na małych porównywalnych korpusach tekstowych,
  - zanurzenia generowane przez dostępne modele oszacowane na dużych jednojęzycznych lub wielojęzycznych korpusach.

Maksymalne/uśrednione zanurzenia zdaniowe

- Zanurzenia zdaniowe powstają poprzez uśrednienie elementów (ang. mean-pooling) lub wybranie maksimów (ang. max-pooling) poszczególnych zanurzeń słownych.
- Testowane (kontekstowe) zanurzenia słowne:
  - Zanurzenia **FastText**[5] [Grave et al., 2018],
  - Zanurzenia **Bert**[6] [Devlin et al., 2018] (z przedostatniej warstwy dostępnego enkodera wielojęzycznego modelu językowego),
  - Zanurzenia **Combo** (z warstwy biLSTM ekstrahującej cechy globalne),
  - (Zanurzenia **BPEmb**[7] [Heinzerling i Strube, 2018] miały porównywalną albo gorszą jakość).

[5] `https://fasttext.cc`

[6] `https://storage.googleapis.com/bert_models/2018_11_23/multi_cased_L-12_H-768_A-12.zip`

[7] `https://nlp.h-its.org/bpemb`

- **Sent2Vec$_{NS}$** – zanurzenia oszacowane na 3M zdań przy użyciu biblioteki Sent2Vec.
- Biblioteka Sent2Vec[8] [Pagliardini et al., 2018]:
  - architektura neuronowa przypomina CBOW [Mikolov et al., 2013],
  - modele są liczone na unigramach i bigramach,
  - funkcja straty liczona z wykorzystaniem techniki *negative sampling*.
- Dane tekstowe (3M zdań) z korpusu *Paralela* [Pęzik, 2016]:
  - bez zdań, które zostały wybrane do danych do testów próbkowania,
  - tylko przyporządkowania typu 1-do-1.

[8] `https://github.com/epfml/sent2vec`

Zanurzenia generowane przez dostępne modele

- **Sent2Vec$_{\text{orig}}$**[9] dla angielskiego:
  - zanurzenia 700-wymiarowe,
  - model trenowany na korpusie Toronto Books[10] (70M zdań).
- **Use**[11] [Cer et al., 2018] dla angielskiego:
  - zanurzenia 512-wymiarowe,
  - model liczony techniką trenowania wielozadaniowego (ang. multi-task learning),
  - model trenowany na danych Wikipedii, e-gazetach, portalach z dialogami, forach dyskusyjnych i korpusie SNLI [Bowman et al., 2015].
- **Laser**[12] [Artetxe i Schwenk, 2018] dla wielu języków:
  - zanurzenia 1024-wymiarowe,
  - model trenowany na 223M zdań równoległych w 93 językach z różnych źródeł.

[9] https://drive.google.com/file/d/0B6VhzidiLvjSdENLSEhrdWprQ0k

[10] http://www.cs.toronto.edu/~mbweb/

[11] https://tfhub.dev/google/universal-sentence-encoder-large/3

[12] https://github.com/facebookresearch/LASER

Spis treści

1. Wprowadzenie
2. Metodologia badawcza
3. Eksperymenty
4. Wyniki
5. Podsumowanie

Wyniki w zadaniach typu „probing” i „downstream”

	FASTTEXT$_{MAX}$	FASTTEXT$_{MEAN}$	BERT$_{MAX}$	BERT$_{MEAN}$	COMBO$_{MAX}$	COMBO$_{MEAN}$	SENT2VEC$_{NS}$	SENT2VEC$_{ORIG}$	LASER	USE
SentLen	52.55	72.27	72.66	82.13	85.03	**87.38**	71.56	64.76	85.98	60.00
	52.63	67.44	70.79	82.19	84.46	86.31	65.15		**86.73**	
WC	24.44	46.73	35.24	45.53	9.39	11.05	59.96	**79.23**	59.79	43.11
	19.83	45.84	38.56	43.60	23.04	26.23	**63.85**		49.03	

Accuracy w %.

Legenda: ANGIELSKI, POLSKI, NAJWYŻSZY WYNIK.

	FASTTEXT$_{MAX}$	FASTTEXT$_{MEAN}$	BERT$_{MAX}$	BERT$_{MEAN}$	COMBO$_{MAX}$	COMBO$_{MEAN}$	SENT2VEC$_{NS}$	SENT2VEC$_{ORIG}$	LASER	USE
TreeDepth	29.91	33.00	33.97	38.20	49.08	**51.87**	33.92	31.03	39.48	31.09
	26.99	30.12	34.43	37.81	44.96	**47.35**	32.84		40.04	
TopDeps	60.49	71.11	78.20	79.33	**93.99**	93.87	75.77	65.31	83.33	63.88
	65.45	70.67	71.68	75.28	88.16	**88.53**	73.44		78.84	
Passive	84.13	89.47	89.77	92.40	**98.48**	98.41	88.73	89.04	92.85	86.61
	85.19	91.92	92.16	94.77	98.41	**98.71**	92.44		95.37	
Tense	75.04	84.47	89.32	90.89	**96.65**	96.64	83.19	85.25	92.19	85.64
	81.56	88.89	93.73	96.09	97.35	**97.47**	87.36		96.87	
SubjNum	73.87	81.43	88.43	90.75	93.19	93.37	82.27	80.88	**94.21**	81.65
	76.73	87.01	89.89	91.51	94.20	**95.03**	87.84		93.79	
ObjNum	71.75	79.24	85.16	86.89	93.23	**94.71**	77.23	80.12	89.33	79.61
	69.41	76.05	80.24	82.64	90.27	**90.31**	74.77		82.53	
SentType	96.23	96.20	97.39	97.76	96.85	96.04	97.17	93.76	**97.84**	85.25
	90.61	96.09	98.36	**98.57**	98.53	98.56	98.09		98.39	

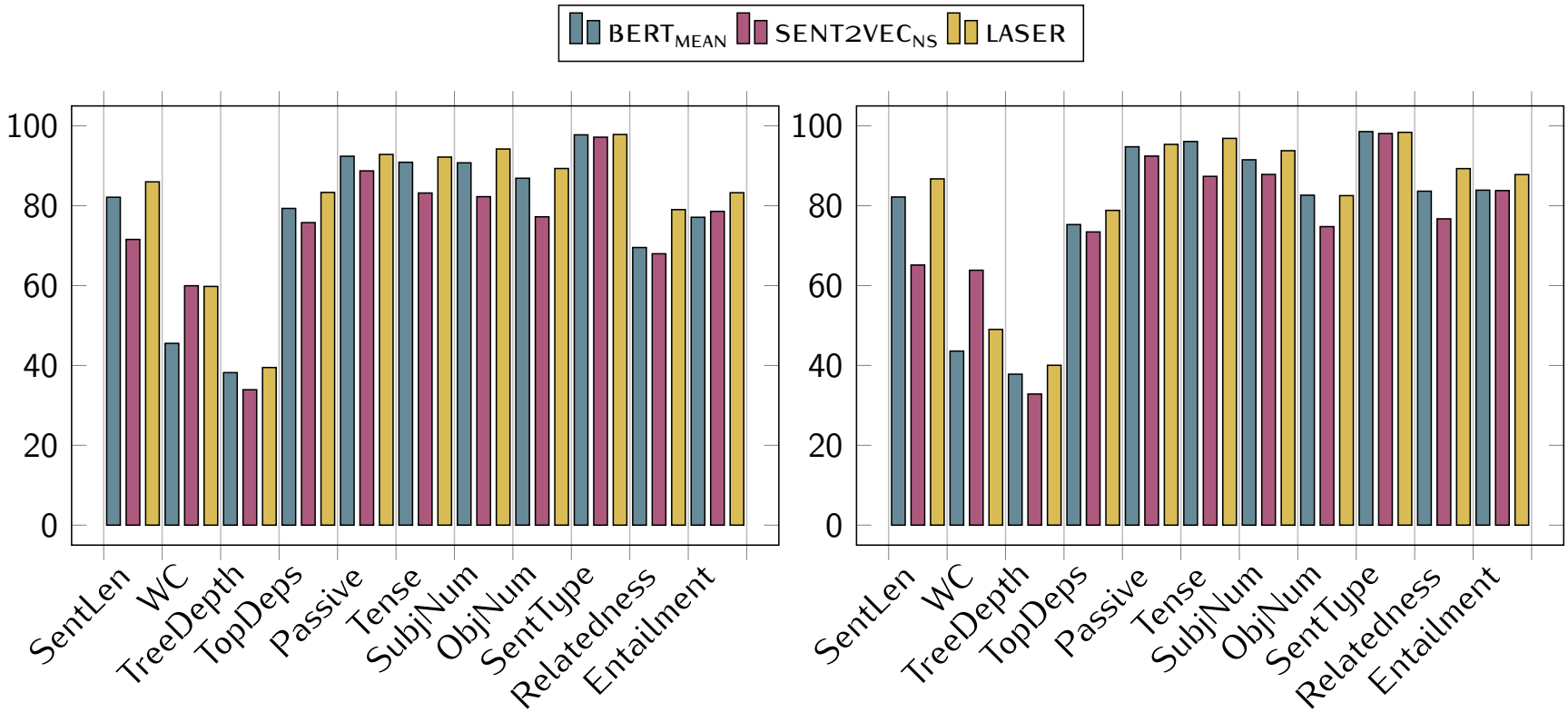
	measure	FASTTEXT$_{MAX}$	FASTTEXT$_{MEAN}$	BERT$_{MAX}$	BERT$_{MEAN}$	COMBO$_{MAX}$	COMBO$_{MEAN}$	SENT2VEC$_{NS}$	SENT2VEC$_{ORIG}$	LASER	USE
Relatedness	r	75.71	76.02	74.23	76.54	58.94	59.38	73.43	79.81	84.54	**86.86**
	ρ	69.35	69.20	68.61	69.54	58.35	58.59	67.97	70.64	79.03	**80.80**
Entailment	r	76.10	78.06	78.46	83.08	77.40	77.44	76.53		**88.09**	
	ρ	77.01	79.31	78.91	83.65	77.81	77.98	76.72		**89.30**	
	a	76.72	76.86	77.71	77.11	72.82	72.58	78.59	78.26	**83.26**	81.77
	a	86.10	87.40	86.70	83.90	84.70	86.10	83.80		**87.80**	

Wszystkie wyniki są wyrażone w %.
Powiązanie semantyczne: współczynniki Pearsona r i Spearmana ρ.
Wynikanie tekstowe: accuracy (a).

Wyniki: uwagi i komentarze

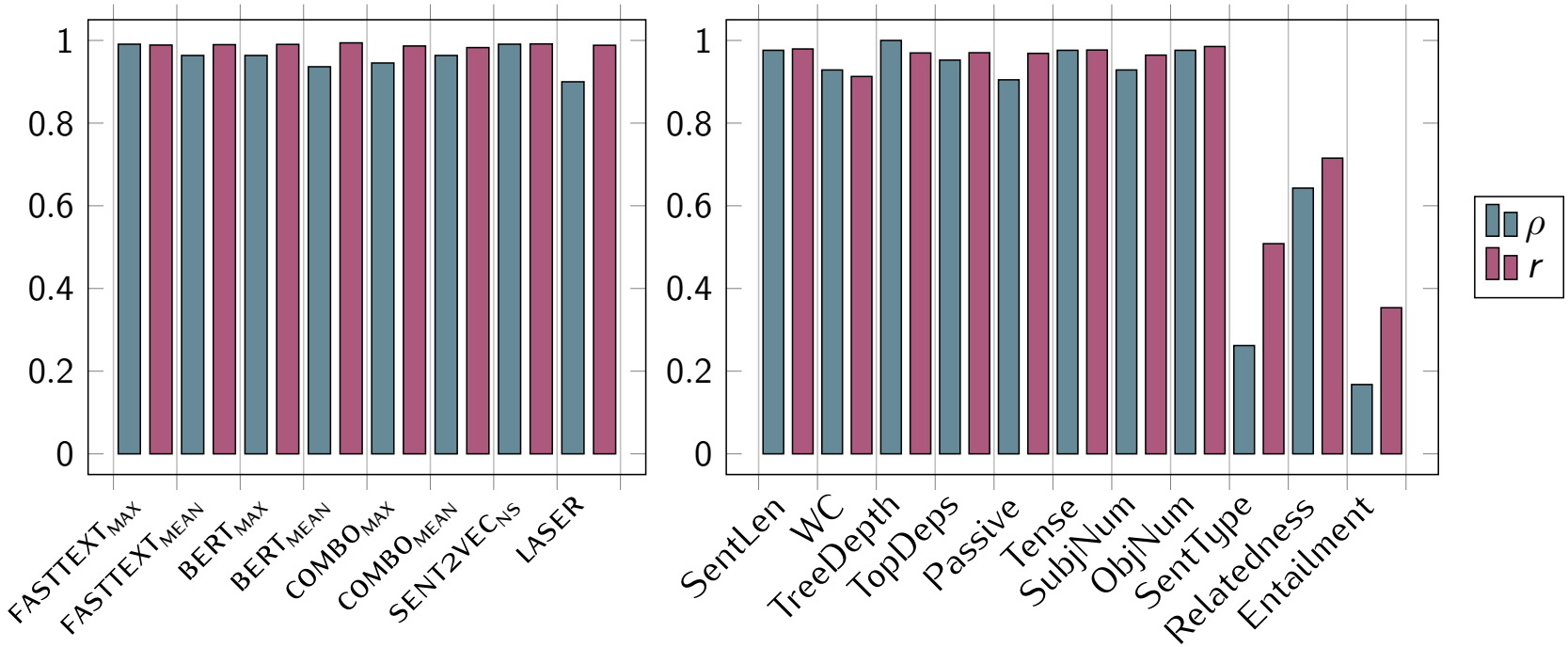
- Dla większości zadań typu „probing” najlepsze wyniki uzyskano dla wektorów COMBO.
- Wektory COMBO zostały wytrenowane w kontekście składniowym (parsowanie zależnościowe), parser COMBO był też używany do generowania danych typu „probing”.
- Pomijając wektory COMBO, najlepsze wyniki w większości testów dały reprezentacje LASER.
- Wektory uśrednione dość konsekwentnie dają lepsze wyniki, niż ich odpowiedniki uzyskane przez *max-pooling*.

Wyniki w każdym zadaniu (wybrane modele)



Lewy: angielski, prawy: polski.
Miary: ρ Spearmana dla Relatedness i *accuracy* dla pozostałych zadań.

Korelacja pomiędzy wynikami dla obu języków



- większość korelacji jest wysoka,
- **SentType**: wszystkie wyniki mieściły się w niewielkim przedziale,
- dane dla zadań typu „downstream” były tworzone niezależnie i mogą być mniej porównywalne.

Spis treści

1. Wprowadzenie
2. Metodologia badawcza
3. Eksperymenty
4. Wyniki
5. Podsumowanie

Powiązania badawcze

- Badania oparte na testach próbkowania na gruncie języka angielskiego i drzew składnikowych zostały zainicjowane przez [Shi et al., 2016] i [Adi et al., 2017], a następnie kontynuowane przez [Conneau et al., 2018].
- Nasze badania stanowią kontynuację tego nurtu badawczego, ale:
  - metoda testów próbkowania została po raz pierwszy użyta w odniesieniu do języka innego niż angielski,
  - testy są oparte na drzewach zależnościowych UD, dlatego są uniwersalne.

Podsumowanie

- Zaprojektowano niezależne od języka testy próbkowania.
- Proponowana procedura ekstrakcji zbiorów danych jest uniwersalna dla wszystkich języków z bankiem drzew UD.
- Udostępniono zbiory danych do testów próbkowania dla angielskiego i polskiego.
- Wnioski:
  - Zanurzenia zdaniowe szacowane przez parser COMBO kodują informacje lingwistyczne w najbardziej trafny sposób.
  - Informacje lingwistyczne są dobrze kodowane w wielojęzycznych zanurzeniach zdań typu LASER.

Podziękowania

Przedstawione badania były finansowane przez Narodowe Centrum Nauki (grant SONATA 8 nr 2014/15/D/HS2/03486). Obliczenia były prowadzone w Poznańskim Centrum Superkomputerowo-Sieciowym.

Bibliografia (1)

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017.

Fine-grained analysis of sentence embeddings using auxiliary prediction tasks.
In *Proceedings of International Conference on Learning Representations (ICLR 2017)*.

Mikel Artetxe and Holger Schwenk. 2018.

Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond.
CoRR, abs/1812.10464.

Luisa Bentivogli, Raffaella Bernardi, Marco Marelli, Stefano Menini, Marco Baroni, and Roberto Zamparelli. 2014.
SICK through the SemEval Glasses. Lesson learned from the evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment.
Journal of Language Resources and Evaluation, 50:95–124.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015.

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. Association for Computational Linguistics.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018.
Universal Sentence Encoder for English.
In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 169–174. Association for Computational Linguistics.

Alexis Conneau and Douwe Kiela. 2018.

SentEval: An Evaluation Toolkit for Universal Sentence Representations.
In *Proceedings of the 11th International Conference on Language Resources and Evaluation*, pp. 1699–1704.

Bibliografia (2)

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018.

What you can cram into a single \$&!#* vector: Probing sentence embeddings for linguistic properties.
In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pp. 2126–2136. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018.

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
CoRR, abs/1810.04805.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018.

Learning Word Vectors for 157 Languages.
In *Proceedings of the 11th International Conference on Language Resources and Evaluation*, pp. 3483–3487.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013.

Distributed Representations of Words and Phrases and their Compositionality.
In *Neural and Information Processing System (NIPS)*.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan T. McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016.
Universal Dependencies v1: A Multilingual Treebank Collection.
In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pp. 1659–1666.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018.

Unsupervised Learning of Sentence Embeddings Using Compositional n-Gram Features.
In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 528–540. Association for Computational Linguistics.

Piotr Pęzik. 2016.

Exploring Phraseological Equivalence with Paralela.
In *Polish-Language Parallel Corpora*, page 67–81. Instytut Lingwistyki Stosowanej UW, Warsaw.

Bibliografia (3)

Piotr Rybak and Alina Wróblewska. 2018.

Semi-Supervised Neural System for Tagging, Parsing and Lematization.
In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pp. 45–54. Association for Computational Linguistics.

Xing Shi, Inkit Padhi, and Kevin Knight. 2016.

Does String-Based Neural MT Learn Source Syntax?
In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1526–1534. Association for Computational Linguistics.

Milan Straka and Jana Straková. 2017.

Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe.
In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pp. 88–99, Vancouver, Canada. Association for Computational Linguistics.

Benjamin Heinzerling and Michael Strube. 2018.

BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages.
In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, pages 2989–2993. European Language Resource Association.

Alina Wróblewska and Katarzyna Krasnowska-Kieraś. 2017.

Polish evaluation dataset for compositional distributional semantics models.
In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp. 784–792. Association for Computational Linguistics.