Empirical Linguistic Study of Sentence Embeddings

Katarzyna Krasnowska-Kieraś **Alina Wróblewska**
Institute of Computer Science, Polish Academy of Sciences
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
kasia.krasnowska@gmail.com alina@ipipan.waw.pl

Abstract

The purpose of the research is to answer the question whether linguistic information is retained in vector representations of sentences. We introduce a method of analysing the content of sentence embeddings based on universal probing tasks, along with the classification datasets for two contrasting languages. We perform a series of probing and downstream experiments with different types of sentence embeddings, followed by a thorough analysis of the experimental results. Aside from dependency parser-based embeddings, linguistic information is retained best in the recently proposed LASER sentence embeddings.

1 Introduction

Modelling natural language with neural networks has been an extensively researched area for several years now. On the one hand, deep learning enormously reduced the cost of feature engineering. On the other hand, we are largely unaware of features that are used in estimating a neural model and, therefore, kinds of information that a trained neural model relies most heavily on. Since neural network-based models work very well in many NLP tasks and often provide state-of-the-art results, it is extremely interesting and desirable to understand which properties of words, phrases or sentences are retained in their embeddings. An approach to investigate whether linguistic properties of English sentences are encoded in their embeddings is proposed by Shi et al. (2016), Adi et al. (2017), and Conneau et al. (2018). It consists in designing a series of classification problems focusing on linguistic properties of sentences, so called *probing tasks* (Conneau et al., 2018). In a probing task, sentences are labelled according to a particular linguistic property. Given a model that generates an embedding vector for any sentence, the model is applied to the probing sentences. A classifier is then trained with the resulting embeddings as inputs and probing labels as targets. The performance of the resulting classifier is considered a proxy for how well the probing property is retained in the sentence embeddings.

We propose an extension and generalisation of the methodology of the probing tasks-based experiments. First, the current experiments are conducted on two typologically and genetically different languages: English, which is an isolating Germanic language and Polish, which is a fusional Slavic one. Our motivation for conducting experiments on two contrasting languages is as follows. English is undoubtedly the most prominent language with multiple resources and tools. However, English language processing is only a part of NLP in general. Methods designed for English are not guaranteed to be universal. In order to verify whether an NLP algorithm is powerful, it is not enough to evaluate it solely on English. Evaluation on additional languages can shed light on an investigated method. We select Polish as our contrasting language for pragmatic reasons, i.e. there is a Polish dataset – CDSCorpus (Wróblewska and Krasnowska-Kieraś, 2017) – which is comparable to the SICK relatedness/entailment corpus (Bentivogli et al., 2014). Both datasets are used in downstream evaluation.

Second, the designed probing tests are universal for both tested languages. For syntactic processing of both languages, we use the Universal Dependencies schema (UD, Nivre et al., 2016).[1] Since we use automatically parsed UD trees for generating probing datasets, analogous tests can be generated for any language with a UD treebank on which a parser can be trained.

[1]The Universal Dependencies initiative aims at developing a cross-linguistically consistent morphosyntactic annotation schema and at building a large multilingual collection of treebanks annotated according to this schema. It is worth nothing that the UD schema has become the *de facto* standard for syntactic annotation in the recent years.

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 5729–5739
Florence, Italy, July 28 - August 2, 2019.

The contributions of this work are twofold. (1) We introduce a method of analysing the content of sentence embeddings based on universal probing tasks, along with the classification datasets for two contrasting languages. (2) We carry out a series of empirical experiments based on publicly released probing datasets[2] created within the described work and the obtainable downstream task datasets with different types of sentence embeddings, followed by a thorough analysis of the experimental results.

We test sentence embeddings obtained with max-pooling and mean-pooling operations over word embeddings or contextualised word embeddings, sentence embeddings estimated on small corpora, and sentence embeddings estimated on large monolingual or multilingual corpora.

2 Experimental Methodology

The purpose of the research is to answer the question whether linguistic information is retained in vector representations of sentences. Assessment of the linguistic content in sentence embeddings is not a trivial task and we verify whether it is possible with a probing task-based method (see Section 2.1). Probing sentence embeddings for individual linguistic properties do not examine the overall performance of embeddings in composing the meaning of the represented sentence. We therefore provide two downstream tasks for a general evaluation (see Section 2.2).

2.1 Probing Task-based Method

A probing task can be defined as "a classification problem that focuses on simple linguistic properties of sentences" (Conneau et al., 2018). A probing dataset contains the pairs of sentences and their categories. For example, the dataset for the **Passive** probing task (the binary classification) consists of two types of the pairs: ⟨a passive voice sentence, 1⟩ and ⟨a non-passive (active) voice sentence, 0⟩. The sentence–category pairs are automatically extracted from a corpus of dependency parsed sentences. The extraction procedure is based on a set of rules compatible with the Universal Dependencies annotation schema. The proposed rules of creating the probing task datasets are thus universal for languages with the UD style dependency treebanks.

A classifier is trained and tested on vector representations of the probing sentences generated with a sentence embedding model. If a linguistic property is encoded in the sentence embeddings and the classifier learns how this property is encoded, it will correctly classify the test sentence embeddings. The efficiency of the classifiers for each probing task is measured with accuracy. The probing tasks are described in Section 3.

2.2 Downstream Task-based Method

Two downstream tasks are proposed in our experiments: **Relatedness** and **Entailment**. The semantic relatedness[3] task is to measure the degree of any kind of lexical or functional association between two terms, phrases or sentences. The efficiency of the classifier for semantic relatedness is measured with Pearson's r and Spearman's ρ coefficients. The textual entailment task is to assess whether the meaning of one sentence is entailed by the meaning of another sentence. There are three entailment classes: *entailment*, *contradiction*, and *neutral*. The efficiency of the classifier for entailment, in turn, is measured with accuracy.

3 Probing Tasks

The point of reference for designing our probing tasks is the work by Conneau et al. (2018). The authors propose several probing tasks and divide them into those pertaining to surface, syntactic and semantic phenomena. However, we decide to discard the 'syntactic versus semantic' distinction and consider all tasks either surface (see Section 3.1) or compositional (see Section 3.2).

This decision is motivated by the fact that both syntactic and semantic principles are undoubtedly compositional by their nature. The syntax admitting well-formed expressions on the basis of the lexicon works in tandem with the semantics. According to Jacobson's notion of *Direct Compositionality* (Jacobson, 2014, 43), "each syntactic rule which predicts the existence of some well-formed expression (as output) is paired with a semantic rule which gives the meaning of the output expression in terms of the meaning(s) of the input expressions".

3.1 Tests on Surface Properties

The tests investigate whether surface properties of sentences (i.e. sentence length and lexical content)

[2] http://git.nlp.ipipan.waw.pl/Scwad/SCWAD-probing-data

[3] Semantic relatedness is not equivalent to semantic similarity. Semantic similarity is only a special case of semantic relatedness, e.g. CAR and AUTO are similar terms and CAR and GARAGE are related terms.

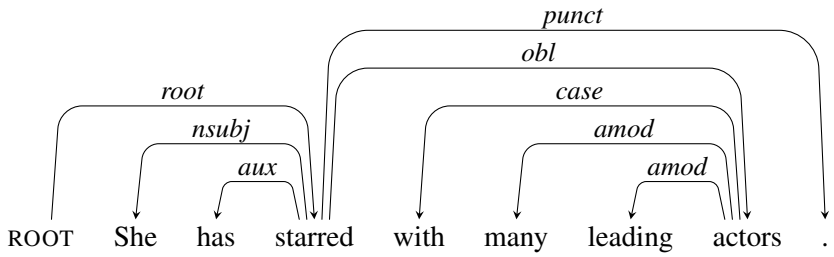

Figure 1: An example UD tree of the sentence *She has starred with many leading actors*.

are retained in their embeddings. We follow the definition of surface probing tasks and the procedure of preparing training data as described by Conneau et al. (2018).

SentLen (sentence length) This task consists in classifying sentences by their length. There are 6 sentence length classes with the following token intervals: 0: (3, 5), 1: (6, 8), 2: (9, 11), 3: (12, 14), 4: (15, 17), 5: (18, 20), 6: (21, 23).
Example: The sentence from Figure 1 has the category 1, since it contains 8 tokens.

WC (word content) This task consists in a 750-way classification of sentences containing exactly one of pre-selected 750 target words (i.e. the categories correspond to the 750 words). The words are selected based on their frequency ranking in the corpus from which the probing datasets were extracted: top 2000 words are discarded and the next 750 words are used as task categories.[4]

3.2 Compositional Tests

The tests on compositional principles are significantly modified (e.g. **TreeDepth**, **TopDeps**, **Tense**) with respect to Conneau et al. (2018) or designed anew (i.e. **Passive** and **SentType**), because the basis for preparing probing datasets is constituted by dependency trees.[5]

TreeDepth (dependency tree depth) This task consists in classifying sentences based on the depth of the corresponding dependency trees. The task is defined similarly to Conneau et al. (2018), but dependency trees are used instead of constituent trees. Similarly to the original TreeDepth task, the data is decorrelated with respect to sentence length.
Example: The dependency tree in Figure 1 has a depth of 3, because the path from the root node to any token node contains 3 tokens at most.

TopDeps (top dependency schema) The idea of this task is based on TopConst task[6] (Conneau et al., 2018), but adapted to dependency trees. The task consists in predicting a multiset of the dependency types labelling the relations between the top-most node (the ROOT's only dependent) and all its children, barring punct relations. The position of a phrase in an English sentence largely determines its grammatical function. In Polish, in turn, word order is relatively free and therefore not a strong determinant of grammatical functions. We thus extract multisets of dependency types, not taking into account the text order of their respective phrases. The extracted multisets roughly correspond to predicate-argument structures. There are 20 classes for each language: 19 most common top dependency schemata and the class {OTHER}.
Example: The **TopDeps** class of the sentence in Figure 1 is {aux nsubj obl}.

Passive (passive voice) This is a binary classification task where the goal is to predict whether a sentence embedding represents a passive voice sentence (the class 1) or an active sentence (the class

[4] Conneau et al. (2018) use 1000 target words selected in a similar manner, but since our datasets are smaller, we proportionally decreased this number in order to maintain the same number of training/validation/testing instances per target word.

[5] We reject the bigram shift task (BShift) as it is applicable only for isolating languages and practically useless for fusional languages with relatively free word order. This task consists in detecting sentences with two random, adjacent words switched. According to Conneau et al. (2018), such shift generally leads to an erroneous utterance (acceptable sentences can be generated accidentally). However, given a language with less strict word order, the intuition is that the BShift procedure could produce too many correct sentences. A very preliminary case study involving several shift strategies and one sentence (*Autorka we wszystkich książkach każe bohaterom szukać tożsamości.* 'The author tells the characters in her all books to look for identity.', lit. 'The author in her all books tells the characters to look for identity.') confirmed this intuition, as most of BShift-modified sentences were accepted by Polish speakers.

[6] In the original TopConst task, the classifier learns to detect one of 19 most common top constructions or <OTHER>, e.g. the top construction sequence of the tree for [*Then*][*very dark gray letters on a black screen*][*appeared*][.] consists of four constituent labels: <ADVP NP VP .>.

0). In case of complex sentences only the voice of the matrix (main) clause is detected.[7] In order to identify passive voice sentences, we adhere to the following procedure: the predicate of a passive voice sentence governs an auxiliary verb and the relation is labelled aux:pass. Furthermore, the predicate (part-of-speech VERB or ADJ) has the features Voice=Pass and VerbForm=Part. The dependency nsubj:pass (passive nominal subject) can be helpful, but as the subject may be dropped in Polish, it is not sufficient.
Example: The active voice sentence in Figure 1 is classified as 0.

Tense (grammatical tense) This is a binary classification of sentences by the grammatical tense of their main predicates. The sentence predicates can be marked for the present (the pres class) or past (the past class) grammatical tense. The present tense predicates have the following properties: the UD POS tag VERB and the feature Tense=Pres. The past tense predicates have the following properties: the UD POS tag VERB and the feature Tense=Past.
Example: The sentence in Figure 1 is classified as past.

SubjNum (grammatical number of subjects) In this binary classification task, sentences are classified by the grammatical number of nominal subjects (marked with the UD label nsubj) of main predicates. There are two classes: sing (the UD POS tag NOUN and the feature Number=Sing) and plur (the UD POS tag NOUN and the feature Number=Plur).
Example: The sentence in Figure 1 is categorised as sing.

ObjNum (grammatical number of objects) This binary classification task is analogous to the one above, but this time sentences are classified by the grammatical number of direct objects of main predicates. The classes are again sing to represent the singular nominal objects (the obj label, the NOUN tag, and the feature Number=Sing), and plur for the plural/mass ones (the obj label, the NOUN tag, and the feature Number=Plur).

[7]The sentence *Although the announcement was probably made to show progress in identifying and breaking up terror cells, I* ***don't find*** *the news that the Baathists continue to penetrate the Iraqi government very hopeful.* is classified as 0, even if it contains the passive voice subordinate clause.

SentType (sentence type) This is a new probing task consisting in classifying sentences by their types. There are three classes: inter for interrogatve sentences (e.g. *Do you like him?*), imper for imperative sentences (e.g. *Get out of here!*), and other for declarative sentences (e.g. *He likes her.*) and exclamatory sentences (e.g. *What a liar!*).

4 Experiments

4.1 SentEval Toolkit

We use the SentEval toolkit (Conneau and Kiela, 2018) in our experiments. The toolkit provides utility for testing any vector representation of sentences in probing and downstream scenarios. Given a function f mapping a list of sentences to a list of vectors (serving as an interface to the tested sentence embedding model), a task and a dataset (with sentences or pairs of sentences as input data), SentEval performs evaluation in the context of the task. More specifically, it generates vectors for the dataset sentences using f, trains a classifier with vectors as inputs and task-specific labels as outputs, and evaluates it. Applying an identical evaluation procedure with the same dataset to different sentence embedding models provides the meaningful comparison of the models.

For the purpose of our tests, the probing datasets provided with the toolkit are replaced with our own, the CDS downstream task dataset is added and the SICK dataset is retained. Other SentEval downstream tasks are not used, having no Polish counterparts. In all experiments we use SentEval's Multilayer Perceptron classifier.[8]

4.2 Probing Datasets

For English and Polish, 9 probing datasets are extracted from *Paralela*[9] (Pęzik, 2016), the largest Polish-English parallel corpus with nearly 4M sentence pairs. An important objective is to make the probing datasets in both languages maximally similar. The choice of a parallel corpus as their source allows to draw probing sentences from collections of texts that have analogous distributions of genre, style, sentence complexity etc. Note that we do not extract parallel sentence pairs (sharing common target classes) for individual probing datasets (sentences are often not translated literally), but we construct English and Polish datasets separately.

[8]With parameters as follows: `kfold=10`, `batch_size=128`, `nhid=50`, `optim=adam`, `tenacity=5`, `epoch_size=4`.

[9]http://paralela.clarin-pl.eu

The sentences are tokenised with UDPipe[10] (Straka and Straková, 2017) and POS-tagged and dependency parsed with COMBO[11] (Rybak and Wróblewska, 2018). The UDPipe and COMBO models are trained on the UD English-EWT treebank[12] (Silveira et al., 2014) with 16k trees (254k tokens) and on the Polish PDB-UD treebank[13] (Wróblewska, 2018) with 22k trees (351k tokens). The set of UD-based rules is applied to dependency-parsed sentences to extract the final probing datasets for both languages.

Following Conneau et al. (2018), for the probing tasks constructed by determining selected properties of a certain dependency tree node (e.g. main predicate's tense, direct object's number, etc.), the division into training, validation and test sets ensures that all data instances, where the relevant token of the sentence (target token) bears the same word form, are not distributed into different sets. For example, all **SubjNum** instances, where the subject phrase is headed by the token *cats* (and the plur class is determined based on the features of this token), are assigned into the same set.

For each probing dataset, only relevant sentences are included (sentences with no subject are irrelevant for **SubjNum**, utterances with no main predicate in present/past tense are irrelevant for **Tense** etc.). Moreover, the target tokens are filtered based on their frequency (most and least frequent are discarded) and the number of occurrences of any target token is limited (to prevent the more frequent ones from dominating the datasets). Finally, the datasets are balanced with relation to the target class.

With the above restrictions implemented, we are able to extract datasets consisting of 90k examples each (75k for training, 7.5k for validation and testing). The dataset sizes are smaller than 120k examples proposed by Conneau et al. (2018), but remain in the same order of magnitude. The lower number of examples per dataset is due to the fact that we strive to build comparable datasets for both investigated languages based on the parallel corpus.

4.3 Downstream Datasets

Two datasets for evaluation of compositional distributional semantic models are used in our experiments. The SICK corpus[14] (Bentivogli et al., 2014) consists of 10k pairs of English sentences. Each sentence pair is human-annotated for relatedness in meaning and entailment. The relatedness score indicates the extent to which meanings of two sentences are related and is calculated as the average of ten human ratings collected for this sentence pair on the 5-point Likert scale. The entailment relation between two sentences, in turn, is labelled with *entailment*, *contradiction*, or *neutral*, selected by the majority of human annotators.

CDSCorpus[15] (Wróblewska and Krasnowska-Kieraś, 2017) is a comparable corpus of 10k pairs of Polish sentences human-annotated for relatedness and entailment. The degree of semantic relatedness between two sentences is calculated as the average of six human ratings on the 0-5-point scale. As an entailment relation between two sentences doesn't have to be symmetric, sentence pairs are annotated with bi-directional entailment labels, i.e. pairs of *entailment*, *contradiction*, and *neutral*.

4.4 Sentence Embeddings

Three types of sentence embeddings are tested in our experiments: (1) sentence embeddings obtained with max-pooling and mean-pooling over pre-trained word embeddings or contextualised word embeddings, (2) sentence embeddings estimated on small comparable corpora, and (3) pre-trained sentence embeddings estimated on large monolingual or multilingual corpora.

Max/Mean-pool Sentence Embeddings Words can be represented as continuous vectors in a low-dimensional space, i.e. word embeddings. Word embeddings are assumed to capture linguistic (e.g. morphological, syntactic, semantic) properties of words. Recently, they are often learnt as part of a neural network trained on an unsupervised or semi-supervised objective task using massive amounts of data (e.g. Mikolov et al., 2013; Grave et al., 2018).[16]

In our experiments, we test FASTTEXT embeddings[17] (Grave et al., 2018) and contextualised word embeddings provided with the multi-layer

[10] https://github.com/ufal/udpipe/releases/tag/v1.2.0

[11] https://github.com/360er0/COMBO

[12] https://github.com/UniversalDependencies/UD_English-EWT

[13] http://git.nlp.ipipan.waw.pl/alina/PDBUD

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

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

[16] Embeddings can also be estimated by dimensionality reduction on a co-occurrence counts matrix (e.g. Pennington et al., 2014).

[17] Pre-trained models from https://fasttext.cc.

bidirectional transformer encoder BERT[18] (Devlin et al., 2018) for English and Polish.[19] Apart from the FASTTEXT and BERT models, we use parts of the dependency parsing models of COMBO to generate sentence embeddings. COMBO has a BiLSTM-based module that produces contextualised word embeddings based on concatenations of word level embeddings and character level embeddings. As the contextualised word embeddings are originally used to predict dependency trees, they should be linguistic information-rich. Since there is some overlap between the PDB-UD treebank (used to train COMBO parsing model for Polish) and CDSCorpus (source of downstream datasets for Polish), a separate COMBO model[20] is trained on PDB-UD data without the overlapping sentences. The model is used to obtain the embeddings for both probing and downstream evaluations.[21]

For all three models listed above, sentence embeddings are obtained by mean or max pooling over individual word embeddings. For FASTTEXT and COMBO, the UDPipe tokenisation of the probing sentences is used and a sequence of embedding vectors is obtained by model lookup and reading the outputs of the parser's BiLSTM module respectively. In the case of BERT (which uses its own tokenisation mechanism), whole sentences are passed to the module and outputs of its penultimate layer are treated as token embeddings.

Small Corpora-based Sentence Embeddings English and Polish sentence embeddings are estimated on *Paralela* corpus. The sentences that are included in any probing dataset are to be excluded from any data used for training sentence embeddings. Furthermore, *Paralela* corpus contains not only 1-to-1 sentence alignments, but also 1-to-many or even many-to-many. As we aim at estimating sentence embedding models, only proper sentences are selected from the corpus. English and Polish sentence embedding models are trained on 3M sentences with the SENT2VEC library[22] (Pagliardini et al., 2018). The SENT2VEC models are estimated with a neural architecture which resembles the CBOW model architecture by Mikolov et al. (2013). The tested models ($\text{SENT2VEC}_{\text{NS}}$) are estimated on unigrams and bigrams with the loss function coupled with negative sampling, to improve training efficiency.

Pre-trained Sentence Embeddings We test English sentence embeddings provided by the pre-trained SENT2VEC and USE models, and multilingual sentence embeddings generated by the LASER model.

The $\text{SENT2VEC}_{\text{ORIG}}$ model[23] trained on the Toronto Book corpus[24] (70M sentences) outputs 700-dimensional sentence embeddings. The Universal Sentence Encoder model[25] (USE, Cer et al., 2018) was estimated in a multi-task learning scenario on a variety of data sources[26] with a Transformer encoder. It takes a variable length English text (e.g. sentence, phrase, or short paragraph) as input and produces a 512-dimensional vector. The Language-Agnostic SEntence Representations model[27] (LASER, Artetxe and Schwenk, 2018) was trained on 223M parallel sentences (93 languages) from various sources. The encoder is implemented as a 5-layer BiLSTM network that represents a sentence as a 1,024-dimensional vector (max-pooling over the last hidden states of the BiLSTM).

5 Results

Results reported by SentEval are summarised in Table 1. The best result for each task in each language is highlighted in grey. For almost all probing tasks, the most accurate embedding is one of the two COMBO-based representations. This is not surprising as the contextualised vector representations produced by COMBO are learnt in the context of dependency parsing. Moreover, the target classes in the probing tasks are derived from trees produced by a parser that uses virtually the same neural model, which can be considered a kind of

[18] Pre-trained language model from https://storage.googleapis.com/bert_models/2018_11_23/multi_cased_L-12_H-768_A-12.zip.

[19] We also tested *BPEmb* embeddings (Heinzerling and Strube, 2018) from https://nlp.h-its.org/bpemb. Sentence embeddings estimated on these word embeddings were of a comparable or worse quality, so we do not give the results.

[20] http://mozart.ipipan.waw.pl/~alina/Polish_dependency_parsing_models/190520_COMBO_PDBUD_noCDS_nosem.pkl

[21] This overlap is in fact only relevant for downstream tasks evaluation. Therefore, for creating the probing datasets, a model based on full PDB-UD treebank is used.

[22] https://github.com/epfml/sent2vec

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

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

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

[26] Estimated on Wikipedia, web news, web question-answer pages, discussion forums, and the Stanford Natural Language Inference corpus (SNLI, Bowman et al., 2015).

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

	language	measure	FASTTEXT$_{MAX}$	FASTTEXT$_{MEAN}$	BERT$_{MAX}$	BERT$_{MEAN}$	COMBO$_{MAX}$	COMBO$_{MEAN}$	SENT2VEC$_{NS}$	SENT2VEC$_{ORIG}$	LASER	USE
SentLen	**E**	a	52.55	72.27	72.66	82.13	85.03	87.38	71.56	64.76	**85.98**	60.00
	P	a	52.63	67.44	70.79	82.19	84.46	86.31	65.15	—	**86.73**	—
WC	**E**	a	24.44	46.73	35.24	45.53	9.39	11.05	59.96	**79.23**	59.79	43.11
	P	a	19.83	45.84	38.56	43.60	23.04	26.23	**63.85**	—	49.03	—
TreeDepth	**E**	a	29.91	33.00	33.97	38.20	49.08	51.87	33.92	31.03	**39.48**	31.09
	P	a	26.99	30.12	34.43	37.81	44.96	47.35	32.84	—	**40.04**	—
TopDeps	**E**	a	60.49	71.11	78.20	79.33	93.99	93.87	75.77	65.31	**83.33**	63.88
	P	a	65.45	70.67	71.68	75.28	88.16	88.53	73.44	—	**78.84**	—
Passive	**E**	a	84.13	89.47	89.77	92.40	98.48	98.41	88.73	89.04	**92.85**	86.61
	P	a	85.19	91.92	92.16	94.77	98.41	98.71	92.44	—	**95.37**	—
Tense	**E**	a	75.04	84.47	89.32	90.89	96.65	96.64	83.19	85.25	**92.19**	85.64
	P	a	81.56	88.89	93.73	96.09	97.35	97.47	87.36	—	**96.87**	—
SubjNum	**E**	a	73.87	81.43	88.43	90.75	93.19	93.37	82.27	80.88	**94.21**	81.65
	P	a	76.73	87.01	89.89	91.51	94.20	95.03	87.84	—	**93.79**	—
ObjNum	**E**	a	71.75	79.24	85.16	86.89	93.23	94.71	77.23	80.12	**89.33**	79.61
	P	a	69.41	76.05	80.24	**82.64**	90.27	90.31	74.77	—	82.53	—
SentType	**E**	a	96.23	96.20	97.39	97.76	96.85	96.04	97.17	93.76	**97.84**	85.25
	P	a	90.61	96.09	98.36	**98.57**	98.53	98.56	98.09	—	98.39	—
Relatedness	**E**	p	75.71	76.02	74.23	76.54	58.94	59.38	73.43	79.81	84.54	**86.86**
		s	69.35	69.20	68.61	69.54	58.35	58.59	67.97	70.64	79.03	**80.80**
	P	p	76.10	78.06	78.46	83.08	77.40	77.44	76.53	—	**88.09**	—
		s	77.01	79.31	78.91	83.65	77.81	77.98	76.72	—	**89.30**	—
Entailment	**E**	a	76.72	76.86	77.71	77.11	72.82	72.58	78.59	78.26	**83.26**	81.77
	P	a	86.10	87.40	86.70	83.90	84.70	86.10	83.80	—	**87.80**	—

Table 1: Probing and downstream task results. Languages: **P**=Polish, **E**=English, measures: a=accuracy, p=Pearson's r, s=Spearman's ρ. All measures are expressed in %.

information leak.

With COMBO models excluded from the ranking due to their obvious handicap, the best-performing sentence embeddings (shown in boldface) for 17 task-language pairs in 22 are yielded by LASER. The exceptions are **ObjNum** and **SentType** for Polish (where the advantage of BERT$_{MEAN}$ is so small it might be insignificant), **Relatedness** for English (suggesting that a comparable USE model could beat LASER in the Polish version of the task as well) and **WC** (where SENT2VEC performs visibly better than all other, even if it is trained on a relatively small corpus).

An interesting observation is that among the pooled embeddings, the MEAN variants quite consistently outperform their MAX counterparts.

Figure 2 visualises the results yielded by selected models in the particular tasks. The models shown are BERT$_{MEAN}$ (the best pooled model), SENT2VEC$_{NS}$ (trained on *Paralela* corpus) and LASER (best-performing apart from COMBO, pre-trained on massive multilingual data). The plots are very similar in shape, the only striking difference being the discrepancy in **WC** results, with LASER and SENT2VEC$_{NS}$ faring similarly (and better than BERT$_{MEAN}$) for English and SENT2VEC$_{NS}$ yielding visibly best results for Polish.

We also measure the correlations between results for Polish and English in two ways. First, for each embedding model we compare the results it yielded in all Polish tasks and all English tasks. Second, for each task type we compare the results obtained using all models in the Polish and English variant of the task.[28] The corresponding correlation coefficients are plotted in Figure 3.

All the per-model correlations are high, which strongly suggests that given embeddings encode a given property similarly well (or poorly) relative to other properties regardless of the language. In the case of per-task correlations, there are three

[28] SENT2VEC$_{ORIG}$ and USE models are excluded from both calculations as they were only tested for English.

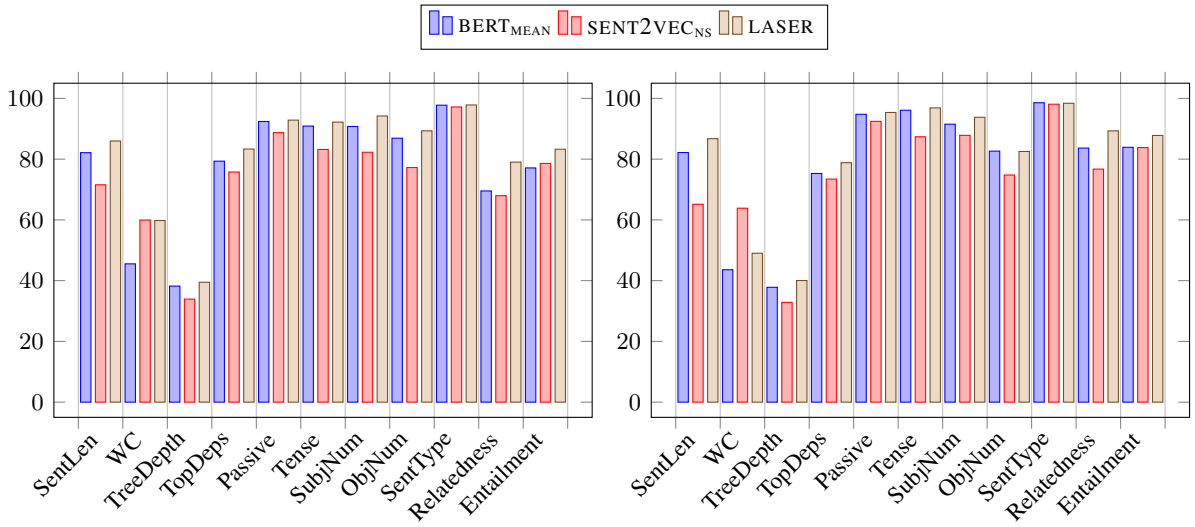

Figure 2: Results in probing and downstream tasks for 3 selected embedding models (left: English, right: Polish). The measure is accuracy (except for Relatedness, where Spearman's ρ is shown). All measures are expressed in %.

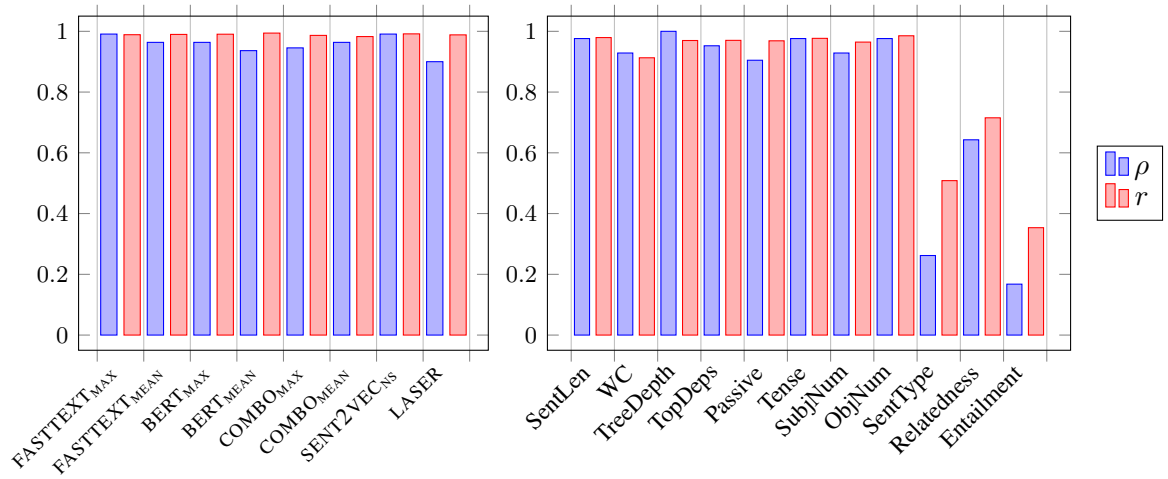

Figure 3: Correlation (measured by Spearman's ρ and Pearson's r) between results for Polish and English (left: per model, right: per task).

tasks with visibly lower correlations: **SentType** and the two downstream tasks. Therefore, for these tasks, the relative performance of individual models differs more between languages. For the downstream tasks this might be partially due to the fact that their respective datasets were created entirely independently and are expected to differ more. As far as **SentType** is concerned, the accuracies obtained for this task are generally very high and most of them fit within a small range.

6 Related Work

Our study follows a research trend in exploring sentence embeddings by means of probing methods, initiated by Shi et al. (2016) and Adi et al. (2017), and continued by Conneau et al. (2018). Investigating NMT systems, Shi et al. (2016) found out that LSTM-based encoders can learn source-language syntax storing different syntactic properties (e.g. voice, tense, top level constituents, part-of-speech tags) in different layers of NMT models. Adi et al. (2017) designed probing tasks for surface properties of sentences (i.e. sentence length, word content, and word order). Two types of sentence embeddings were tested: averaging of CBOW word embeddings and sentence representation output by a LSTM encoder. Conneau et al. (2018) carried out a series of the large-scale experiments on understanding English sentence embeddings with human-validated upper bounds for all probing tasks. They designed 10 probing tasks capturing simple linguistic properties of sentences, tested various

sentence encoding architectures (i.e. BiLSTM and gated convolutional network), and various training objectives (e.g. neural machine translation, autoencoding, SkipThought). Following the mentioned approaches, we examine how much linguistic information is retained in sentence embeddings using 9 similar probing tasks. However, Universal Dependency trees instead of constituent trees are the core of our probing tasks. Furthermore, our experiments are carried out on two contrasting languages, to verify the validity of the evaluation method proposed for English in another language experimental scenario.

Ettinger et al. (2018) considered a very important aspect of sentence meaning – composition. They proposed a method of assessing compositional meaning content in sentence embeddings on the examples of semantic role and negation phenomena. This study has drawn our attention to the compositional dimension of our probing tasks.

Related works by Linzen et al. (2016) and Warstadt and Bowman (2019) proposed evaluation of sentence encoders (e.g. LSTM, transformers) in terms of their ability to learn grammatical information, e.g. to assess sentences as grammatically correct or not (i.e. acceptability judgments).

Finally, several studies were devoted to exploring morphosyntactic properties of sentence embeddings in neural machine translation systems (e.g. Shi et al., 2016; Belinkov et al., 2017).

7 Conclusion

We presented a methodology of empirical research on retention of linguistic information in sentence embeddings using probing and downstream tasks. In the probing-based scenario, a set of language-independent tests was designed and probing datasets were generated for two contrasting languages – English and Polish. The procedure of generating probing datasets is based on the Universal Dependency schema. It is thereby universal for all languages with a UD treebank on which a natural language pre-processing system can be trained. In the downstream-based scenario, the publicly available datasets for semantic relatedness and entailment were used.

We performed a series of probing and downstream experiments with three types of sentence embeddings in the SentEval environment, followed by a thorough analysis of the linguistic content of sentence embeddings. We found out that the COMBO-based embeddings designed to convey morphosyntax encode linguistic information in the most accurate way. Aside from COMBO embeddings, linguistic information is retained most exactly in the recently proposed LASER sentence embeddings, provided by an encoder designed with a relatively simple BiLSTM architecture, but estimated on tremendous multilingual data. Further research is required to find out in what lies the success of LASER embeddings: in the embedding size, in the magnitude of training data, or maybe in the multitude of used languages.

Acknowledgments

The research presented in this paper was supported by SONATA 8 grant no 2014/15/D/HS2/03486 from the National Science Centre Poland.

References

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.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017. What do Neural Machine Translation Models Learn about Morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 861–872. Association for Computational Linguistics.

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*, pages 169–174. Association for Computational Linguistics.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An Evaluation Toolkit for Universal Sentence Representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, pages 1699–1704. European Language Resource Association.

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 (Volume 1: Long Papers)*, pages 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.

Allyson Ettinger, Ahmed Elgohary, Colin Phillips, and Philip Resnik. 2018. Assessing Composition in Sentence Vector Representations. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1790–1801. Association for Computational Linguistics.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning Word Vectors for 157 Languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, pages 3483–3487. European Language Resource Association.

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.

Pauline Jacobson. 2014. *Compositional Semantics. An Introduction to the Syntax/Semantics Interface*. Oxford Textbooks in Linguistics. Oxford University Press.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

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 Tenth International Conference on Language Resources and Evaluation LREC 2016*, pages 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, Volume 1 (Long Papers)*, pages 528–540. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. 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.

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*, pages 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.

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Christopher D. Manning. 2014. A Gold Standard Dependency Corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 2897–2904. European Language Resource Association.

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*, pages 88–99, Vancouver, Canada. Association for Computational Linguistics.

Alex Warstadt and Samuel R. Bowman. 2019. Grammatical Analysis of Pretrained Sentence Encoders with Acceptability Judgments. *CoRR*, abs/1901.03438.

Alina Wróblewska. 2018. Extended and Enhanced Polish Dependency Bank in Universal Dependencies Format. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 173–182. Association for Computational Linguistics.

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 (Volume 1: Long Papers)*, pages 784–792. Association for Computational Linguistics.