Evaluation of contextual embeddings on less-resourced languages

Matej Ulčar[1], Aleš Žagar[1], Carlos S. Armendariz[2], Andraž Repar[3]
Senja Pollak[3], Matthew Purver[2], Marko Robnik-Šikonja[1]

[1] University of Ljubljana, Faculty of Computer and Information Science
Večna pot 113, 1000 Ljubljana, Slovenia
{matej.ulcar, ales.zagar, marko.robnik}@fri.uni-lj.si

[2] Queen Mary University of London, Cognitive Science Research Group,
Mile End Road, London E1 4NS, United Kingdom
{c.santosarmendariz, m.purver}@qmul.ac.uk

[3] Jožef Stefan Institute,
Jamova 39, 1000 Ljubljana, Slovenia
{andraz.repar, senja.pollak}@ijs.si

Abstract

The current dominance of deep neural networks in natural language processing is based on contextual embeddings such as ELMo, BERT, and BERT derivatives. Most existing work focuses on English; in contrast, we present here the first multilingual empirical comparison of two ELMo and several monolingual and multilingual BERT models using 14 tasks in nine languages. In monolingual settings, our analysis shows that monolingual BERT models generally dominate, with a few exceptions such as the dependency parsing task, where they are not competitive with ELMo models trained on large corpora. In cross-lingual settings, BERT models trained on only a few languages mostly do best, closely followed by massively multilingual BERT models.

1 Introduction

Deep neural networks have dominated the area of natural language processing (NLP) for almost a decade. The establishment of contextual embeddings such as ELMo [Peters et al., 2018] and BERT [Devlin et al., 2019] have advanced many NLP tasks to previously unattainable performance, often achieving human levels. At the same time, newer generations of transformer-based [Vaswani et al.,

2017] neural language models have increased in size and training times to levels unachievable to anyone but large corporations. While training BERT with 110 million parameters is possible on 8 GPUs in about a month, the T5 model [Raffel et al., 2020] contains 11 billion parameters and GPT-3 [Brown et al., 2020] contains 175 billion parameters. These large models are trained on English. With the exception of Chinese, no similar-sized models exist for another language, while several BERT-sized models have sprouted for other languages.

In this paper, we focus on empirical mono- and cross-lingual comparison of ELMo and BERT contextual models for less-resourced but technologically still relatively well-supported European languages. This choice stems from the enabling conditions for such a study: availability of contextual ELMo and BERT models and availability of evaluation datasets. These constraints and limited space have led us to select nine languages (Croatian, English[1], Estonian, Finnish, Latvian, Lithuanian, Russian, Slovene, Swedish) and seven categories of datasets: named-entity recognition (NER), part-of-speech (POS) tagging, dependency parsing (DP), analogies, contextual similarity (CoSimLex), terminology alignment, and the SuperGLUE suite of benchmarks (eight tasks). While some tasks are well-known and frequently used in the presentation of new models, others are used here for the first time in cross-lingual comparisons. We compare two types of ELMo models (described in Section 3.2) and three categories of BERT models: monolingual, massively multilingual and moderately multilingual (trilingual to be precise). The latter models are specifically intended for cross-lingual transfer.

The aim of the study is to compare i) the quality of different monolingual contextual models and ii) the success of cross-lingual transfer between similar languages and from English to less-resourced languages. While partial comparisons exist for individual languages (in particular English) and individual tasks, no systematic study has yet been conducted. This study fills this gap.

The main contributions of the work are as follows.

1. The establishment of a set of mono- and cross-lingual datasets suitable for the evaluation of contextual embeddings in less-resourced languages.

2. The first systematic monolingual evaluation of ELMo and BERT contextual embeddings for a set of less-resourced languages.

3. The first systematic evaluation of cross-lingual transfer using contextual ELMo and BERT models for a set of less-resourced languages.

The structure of the paper is as follows. Section 2 outlines related work. In Section 3, we describe the monolingual and cross-lingual embedding approaches used. We split them into four categories: baseline non-contextual fastText embeddings, contextual ELMo embeddings, cross-lingual maps for these, and BERT-based monolingual and cross-lingual models. In Section 4, we present

---

[1]We included English for comparison with other languages and for cross-lingual knowledge transfer.

our evaluation scenarios, divided into settings and benchmarks. Section 5 contains the results of the evaluations. We first cover the monolingual approaches, followed by the cross-lingual ones. We present our conclusions in Section 6.

2   Related works

Ever since their introduction, ELMo [Peters et al., 2018] and BERT [Devlin et al., 2019] have attracted enormous attention from NLP researchers and practitioners. Rogers et al. [2020] present a recent survey of over 150 papers investigating the information BERT contains, modifications to its training objectives and architecture, its overparameterization, and compression.

At the time of its introduction, ELMo has been shown to outperform previous pretrained word embeddings like word2vec and GloVe on many NLP tasks [Peters et al., 2018]. Later, BERT models turned out to be even more successful on these tasks [Devlin et al., 2019] and many others [Wang et al., 2019a]. This fact would seemingly make ELMo obsolete. However, concerning the quality of extracted vectors, ELMo can be advantageous [Škvorc et al., 2020]. Namely, the information it contains is condensed into only three layers, while multilingual BERT uses 14 layers and the useful information is spread across all of them [Tenney et al., 2019]. For that reason, we empirically evaluate both ELMo and BERT.

There are two works analysing ELMo on several tasks in a systematic way. Both are limited to English. The original ELMo paper [Peters et al., 2018] uses six tasks: question answering, named entity extraction, sentiment analysis, textual entailment, semantic role labelling, and coreference resolution. Introducing the GLUE benchmark, Wang et al. [2019b] analyzed ELMo on nine tasks: linguistic acceptability, movie review sentiment, paraphrasing, question answering (two tasks), text similarity, natural language inference (two tasks), and coreference resolution. For BERT, there are also two systematic empirical evaluations on English. The original BERT paper [Devlin et al., 2019] used nine datasets in the GLUE benchmark and three more tasks: question answering, NER, and sentence completion. The SuperGLUE benchmark [Wang et al., 2019a] contains eight tasks where BERT was tested: four question answering tasks, two natural language inference tasks, coreference resolution, and word-sense disambiguation.

Other works study ELMo and/or BERT for individual tasks like NER [Taillé et al., 2020], dependency parsing [Li et al., 2019], diachronic changes [Rodina et al., 2020], sentiment analysis [Robnik-Šikonja et al., 2021], or coreference resolution [Joshi et al., 2019]. Several papers introduce language specific ELMo or BERT models and evaluate it on tasks available in that language, e.g., Russian [Kuratov and Arkhipov, 2019], French [Martin et al., 2020], or German [Risch et al., 2019]. The instances of these models used in our evaluation are described in Section 3.4.2.

In cross-lingual settings, most works compare massively multilingual BERT models such as mBERT [Devlin et al., 2019] and XLM-R [Conneau et al., 2019]. Ulčar and Robnik-Šikonja [2020] trained several trilingual BERT models suit-

able for transfer from English to less-resourced similar languages. The massively multilingual and trilingual models are described in Section 3.4.1 and Section 3.4.3, respectively.

In contrast to the above studies, our analysis of ELMo and BERT includes nine languages (including English), 14 tasks, and covers both monolingual and cross-lingual evaluation. By translating SuperGLUE tasks into Slovene, we offer the first cross-lingual evaluation of multilingual BERT models in this benchmark suite and also compare human and machine translations.

3 Cross-lingual and contextual embedding

In this section, we briefly describe the monolingual and cross-lingual approaches that we compare. In Section 3.1, we briefly present the non-contextual fastText baseline, and in Section 3.2, the contextual ELMo embeddings. Mapping methods for the explicit embedding spaces produced by these two types of approaches are discussed in Section 3.3. We describe large pretrained language models based on the transformer neural network architecture (i.e. BERT variants) in Section 3.4.

3.1 Baseline fastText embeddings

As deep neural networks became the predominant learning method for text analytics, it was natural that they also gradually became the method of choice for text embeddings. A procedure common to these embeddings is to train a neural network on one or more semantic text classification tasks and then take the weights of the trained neural network as a representation for each text unit (word, n-gram, sentence, or document). The labels required for training such a classifier come from huge corpora of available texts. Typically, they reflect word co-occurrence, like predicting the next or previous word in a sequence or filling in missing words but may be extended with other related tasks, such as sentence entailment. The positive instances for the training are obtained from texts in the used corpora, while the negative instances are mainly obtained with negative sampling (sampling from instances that are highly unlikely related).

Mikolov et al. [2013] introduced the word2vec method and trained it on a huge Google News data set (about 100 billion words). The pretrained 300-dimensional vectors for 3 million English words and phrases are publicly available[2]. Word2vec consists of two related methods, *continuous bag of words (CBOW)* and *skip-gram*. Both methods construct a neural network to classify co-occurring words by taking as an input a word and its d preceding and succeeding words, e.g., $\pm$ 5 words.

Bojanowski et al. [2017] developed the fastText method, built upon the word2vec method but introduced subword information, which is more appropriate for morphologically rich languages such as the ones processed in this work. They took the skip-gram method from word2vec and edited the scoring

---

[2] https://code.google.com/archive/p/word2vec/

function used to calculate the probabilities. In the word2vec method, this scoring function is equal to a dot product between two word vectors. For words w_t and w_c and their respective vectors u_t and u_c, the scoring function s is equal to $s(w_t, w_c) = \mathbf{u}_t^\top \mathbf{u}_c$. The scoring function in fastText is a sum of dot products for each subword (i.e. character n-gram) that appears in the word w_t:

$$s(w_t, w_c) = \sum_{g \in G_t} \mathbf{z}_g^\top \mathbf{u_c},$$

where $\mathbf{z}_g$ is a vector representation of an n-gram (subword) g and G_t is a set of all n-grams (subwords) appearing in w_t. As fastText is conceptually very similar to word2vec, we do not treat them as different methods but only test fastText as the baseline.

3.2 ELMo embeddings

ELMo (Embeddings from Language Models) embedding [Peters et al., 2018] is an example of a pretrained transfer learning model. The first layer is a CNN (Convolutional Neural Network) layer, which operates on a character level. This layer is context-independent, so each word always gets the same embedding, regardless of its context. It is followed by two biLM (bidirectional language model) layers. A biLM layer consists of two concatenated LSTMs [Hochreiter and Schmidhuber, 1997]. The first LSTM predicts the following word, based on the given past words, where each word is represented by the embeddings from the CNN layer. The second LSTM predicts the preceding word based on the given following words. The second LSTM layer is equivalent to the first LSTM, just reading the text in reverse.

The actual embeddings are constructed from the internal states of a bidirectional LSTM neural network. Higher-level layers capture context-dependent aspects, while lower-level layers capture aspects of syntax [Peters et al., 2018]. To train the ELMo network, one inputs one sentence at a time. The representation of each word depends on the whole sentence, i.e. it reflects the contextual features of the input text and thereby polysemy of words. For an explicit word representation, one can use only the top layer. Still, more frequently, one combines all layers into a vector. The representation of a word or a token t_k at position k is composed of

$$R_k = \{x_k^{LM}, \overrightarrow{h}_{k,j}^{LM}, \overleftarrow{h}_{k,j}^{LM} \mid j = 1, \ldots, L\} \tag{1}$$

where L is the number of layers (ELMo uses $L = 2$), index j refers to the level of bidirectional LSTM network, x is the initial token representation (either word or character embedding), and h^{LM} denotes hidden layers of forward or backward language model.

In NLP tasks, any set of these embeddings may be used; however, a weighted average is usually used. The weights of the average are learned during the training of the model for the specific task. Additionally, an entire ELMo model can be fine-tuned on a specific end task.

We compare two variants of ELMo models. The ELMoForManyLangs project (EFML) [Che et al., 2018] trained ELMo models for several languages but used relatively small datasets of 20 million words randomly sampled from the raw text released by the CONLL 2017 shared task (wikidump + common crawl) [Ginter et al., 2017]. We group ELMo models using the same architecture, but trained on much larger datasets (from 270 million to 5,5 billion words) under the name L-ELMo (Large ELMo). In L-ELMo we include the original English 5.5B ELMo model[3], Russian ELMo model trained by DeepPavlov[4] on the Russian WMT News, and the ELMo models trained by the EMBEDDIA project[5] for seven languages (Croatian, Estonian, Finnish, Latvian, Lithuanian, Slovene, and Swedish) [Ulčar and Robnik-Šikonja, 2020].

3.3 Cross-lingual maps for fastText and ELMo

Cross-lingual alignment methods take precomputed word embeddings for each language and align them with the optional use of bilingual dictionaries. The goal of alignments is that the embeddings for words with the same meaning shall be as close as possible in the final vector space. Søgaard et al. [2019] overview the area of cross-lingual embeddings. A comprehensive summary of existing mapping approaches can be found in [Artetxe et al., 2018]. Special cross-lingual mapping techniques for contextual ELMo embeddings are presented in [Ulčar and Robnik-Šikonja, 2021].

Context-dependent embedding models calculate a word embedding for each word's occurrence; thus, a word gets a different vector for each context. Mapping such vector spaces from different languages is not straightforward. Schuster et al. [2019] observed that vectors representing different occurrences of each word form clusters. They averaged the vectors for each word occurrence so that each word was represented with only one vector, a so-called anchor. They applied the same procedure to both languages and aligned the anchors using the supervised or unsupervised method of MUSE [Conneau et al., 2018]. However, this method comes with a loss of information. Many words have multiple meanings, which cannot be averaged. For example, the word *mouse* can mean a small rodent or a computer input device. Context-dependent models correctly assign significantly different vectors to these two meanings since they appear in different contexts. Further, a word in one language can be represented with several different words (one for each meaning) in another language or vice versa. By averaging the contextual embedding vectors, we lose these distinctions in meaning.

To align contextual ELMo embeddings, Ulčar and Robnik-Šikonja [2021] developed four methods that take different contexts and word meanings into account. All methods require a different type of contextual mapping datasets, described below. The datasets map not only words but also their contexts. Two mappings follow the existing mapping techniques used for static embeddings and

---

[3]https://allennlp.org/elmo
[4]https://github.com/deepmipt/DeepPavlov
[5]http://hdl.handle.net/11356/1277

assume that the mapping spaces are isomorphic, while the other two drop this assumption. The isomorphic mapping methods are called Vecmap [Artetxe et al., 2018] and MUSE [Conneau et al., 2018], and two non-isomorphic mapping methods are named ELMoGAN-O and ELMoGAN-10k [Ulčar and Robnik-Šikonja, 2021].

To form a cross-lingual mapping between contextual embeddings, a word in one language has to be represented with several different words (one for each meaning) in another language. For that, we require two resources: a sentence aligned parallel corpus of the two covered languages and their bilingual dictionary. The dictionary alone is not sufficient, as the words are not given in the context. Therefore, we can not use it for the alignment of contextual embeddings. The parallel corpus alone is also not sufficient as the alignment is on the level of paragraphs or sentences and not on the level of words. By combining both resources, we take a translation pair from the dictionary and find sentences in the parallel corpus, with one word from the pair present in the sentence of the first language and the second word from the translation pair present in the second language sentence. As a result, we get matching words in matching contexts (sentences) which can serve as contextual anchor points. The details are presented in [Ulčar and Robnik-Šikonja, 2021].

3.4 BERT embeddings

BERT (Bidirectional Encoder Representations from Transformers) embedding [Devlin et al., 2019] generalises the idea of language models (LM) to masked language models (MLM). The MLM randomly masks some of the tokens from the input, and the task of LM is to predict the missing token based on its neighbourhood. BERT uses the transformer architecture of neural networks [Vaswani et al., 2017] in a bidirectional sense and further introduces the task of predicting whether two sentences appear in a sequence. The input representation of BERT are sequences of tokens representing subword units. The input to the BERT encoder is constructed by summing the embeddings of corresponding tokens, segments, and positions. Some widespread words are kept as single tokens; others are split into subwords (e.g., frequent stems, prefixes, suffixes—if needed, down to single letter tokens). The original BERT project offers pretrained English, Chinese, Spanish, and multilingual models.

To use BERT in classification tasks only requires adding connections between its last hidden layer and new neurons corresponding to the number of classes in the intended task. Then, the fine-tuning process is applied to the whole network; all the parameters of BERT and new class-specific weights are fine-tuned jointly to maximise the log-probability of the correct labels.

BERT has shown excellent performance on 11 NLP tasks: 8 from GLUE language understanding benchmark [Wang et al., 2019b], question answering, named entity recognition, and common-sense inference [Devlin et al., 2019]. The performance on monolingual tasks has often improved upon ELMo. However, while multilingual BERT covers 104 languages, its subword dictionary comprises tokens from all covered languages, which might not be optimal for a particular

language. Further, similarly to ELMo, its training and tuning are computationally highly demanding tasks out of reach for most researchers.

Below we first describe the massively multilingual models, followed by monolingual and trilingual models used in our experiments.

3.4.1 Massively multilingual BERT and RoBERTa models

The multilingual BERT model (mBERT) [Devlin et al., 2019] was trained simultaneously on 104 languages, using the available Wikipedia texts in these languages. The mBERT model provides a representation in which the languages are embedded in the same space without requiring further explicit cross-lingual mapping. This massively multilingual representation might be sub-optimal for any specific language or a subset of languages.

Deriving from BERT, Liu et al. [2019] developed RoBERTa, which drops the sentence inference training task (predicting if two given sentences are consecutive or not) and keeps only masked token prediction. Unlike BERT, which generates masked corpus as a training dataset in advance, RoBERTa randomly masks a given percentage of tokens on the fly. In that way, in each epoch, a different subset of tokens get masked. Conneau et al. [2019] used RoBERTa architecture to train the massive multilingual XLM-RoBERTa (XLM-R) model, using 100 languages, akin to the mBERT model.

3.4.2 Monolingual BERT-like models

Following the success of BERT, similar large pretrained transformer language models appeared in other languages.

Kuratov and Arkhipov [2019] trained a monolingual Russian BERT (RuBERT) on Russian Wikipedia and news corpus. They used multilingual BERT (mBERT) to initialise all the model weights, except for the first layer embeddings, where they replaced the mBERT's vocabulary with Russian-only vocabulary. They offer the model via open source DeepPavlov library[6].

Monolingual Finnish BERT (FinBERT) [Virtanen et al., 2019] model was trained on a 3.3 billion token corpus, composed of news (Yle, STT), online discussions (Suomi24) and internet crawl of Finnish websites. The online discussions part of the corpus represents more than half of the entire training data. FinBERT model shares the architecture with the BERT-base model, with 12 transformer layers and the hidden layer size of 768.

Estonian (EstBERT) [Tanvir et al., 2020], Latvian (LVBERT) [Znotiņš and Barzdiņš, 2020], and Swedish (KB-BERT) [Malmsten et al., 2020] BERT models were all trained in the same manner as FinBERT. Estonian EstBERT was trained on a 1.1 billion word Estonian National Corpus 2017, comprised of Estonian Reference Corpus (90s–2008), Estonian Web (2013 and 2017), and Estonian Wikipedia (2017). Latvian LVBERT was trained on a relatively small corpus with 500 million tokens. It consists mostly of articles and comments from various news portals, while including also Latvian Balanced corpus LVK2018 and

---

[6]https://github.com/deepmipt/DeepPavlov

Latvian Wikipedia. National Library of Sweden (KB) trained KB-BERT on modern Swedish corpora, using resources from 1940 to 2019. The 3.5 billion word corpora are composed mostly of digitised newspapers and include also government publications, Swedish Wikipedia, comments from online forums, etc. The quality of these monolingual BERT models varies, mostly depending on the training datasets size and quality.

Slovene (SloBERTa) and Estonian (Est-RoBERTa) monolingual models were trained on large non-public high-quality datasets within the EMBEDDIA project[7]. Both models closely follow the architecture and training approach of the Camembert base model [Martin et al., 2020], which is itself based on RoBERTa. Both models have 12 transformer layers and approximately 110 million parameters. SloBERTa was trained for 200,000 steps (about 98 epochs) on Slovene corpora, containing 3.47 billion tokens in total. The corpora are composed of general language corpus, web-crawled texts, academic writings (BSc/BA, MSc/MA and PhD theses) and texts from Slovenian parliament. Est-RoBERTa was trained for about 40 epochs on Estonian corpora, containing mostly news articles from Ekspress Meedia, in total 2.51 billion tokens. The subword vocabularies contain 32,000 tokens for SloBERTa model and 40,000 tokens for Est-RoBERTa model. Both models are publicly available via the popular Hugging Face library[8,9] and for individual download from CLARIN[10,11].

BERTić [Ljubešić and Lauc, 2021] is a transformer-based pretrained model using the Electra approach [Clark et al., 2019]. Electra models train a smaller generator model and the main, larger discriminator model whose task is to discriminate whether a specific word is an original word from the text or a word generated by the generator model. The authors claim that the Electra approach is computationally more efficient than the BERT models based on masked language modelling. BERTić is a BERT-base sized model (110 million parameters and 12 transformer layers), trained on crawled texts from the Croatian, Bosnian, Serbian and Montenegrin web domains. While BERTić is a multilingual model, we use it as a monolingual model and apply it to the Croatian language datasets. Two reasons are supporting this decision. First, most training texts are Croatian (5.5 billion words out of 8 billion). Second, the covered South Slavic languages are closely related, mutually intelligible, and are classified under the same HBS (Serbo-Croatian) macro-language by the ISO-693-3 standard.

3.4.3 Trilingual BERT models

While massively multilingual models allow for a good cross-lingual transfer of trained models, they contain a relatively small input dictionary for each language and most of the words are composed of several tokens. A possible solution is to build BERT models on fewer similar languages. Ulčar and Robnik-Šikonja

---

[7]http://www.embeddia.eu
[8]https://huggingface.co/EMBEDDIA/sloberta
[9]https://huggingface.co/EMBEDDIA/est-roberta
[10]http://hdl.handle.net/11356/1397
[11]https://doi.org/10.15155/9-00-0000-0000-0000-00226L

[2020] constructed trilingual models featuring two similar languages and one highly resourced language (English). Because these models are trained on a small number of languages, they better capture each of them and offer better monolingual performance. At the same time, they can be used in a cross-lingual manner for knowledge transfer from a high-resource language to a low-resource language or between similar languages.

We analyze three trilingual models, the first trained on Slovene, Croatian and English data (CroSloEngual BERT), the second on Estonian, Finnish and English (FinEst BERT), and the third on Latvian, Lithuanian and English (LitLat BERT). The models are publicly available via the popular Huggingface library[12,13,14] and for individual download from CLARIN[15,16,17]. Each model was trained on deduplicated corpora from all three languages.

FinEst BERT and CroSloEngual BERT were trained on BERT-base architecture [Ulčar and Robnik-Šikonja, 2020], using bert-vocab-builder[18] to produce wordpiece vocabularies (composed of subword tokens) from the given corpora. The created wordpiece vocabularies contain 74,986 tokens for FinEst and 49,601 tokens for the CroSloEngual model. The training dataset is a masked corpus. The training randomly masked 15% of the tokens in the corpus and repeated the process five times, each time masking different 15% of the tokens. The dataset is thus five times larger than the original corpora. On this data, trilingual BERT models were trained for about 40 epochs, which is approximately the same as multilingual BERT.

Later LitLat BERT is based on the RoBERTa architecture, which has proven more robust and better performing than BERT. It offers two practical benefits over the original BERT approach. By dropping the next-sentence prediction training task, corpora shuffled on the sentence level can be used in training at the expense of more limited context (compared to the original 512 tokens used in BERT). The second benefit is that it allows for training on multiple GPUs out of the box, while BERT can only be trained on a single GPU unless complex workarounds are implemented. In training, the Lithuanian, Latvian and English corpora were split into three sets, train, eval and test. Train dataset contains 99% of all the corpora; the other two sets contain 0.5% each. The sentencepiece algorithm[19] produced subword byte-pair-encodings (BPE) from a given train dataset. The created subword vocabulary contained 84,200 tokens. The model was trained for 40 epochs, with a maximum sequence length of 512 tokens. Like with FinEst BERT and CroSloEngual BERT, 15% of the tokens were randomly masked during the training.

---

[12] https://huggingface.co/EMBEDDIA/crosloengual-bert
[13] https://huggingface.co/EMBEDDIA/finest-bert
[14] https://huggingface.co/EMBEDDIA/litlat-bert
[15] http://hdl.handle.net/11356/1317
[16] https://doi.org/10.15155/9-00-0000-0000-0000-0021CL
[17] http://hdl.handle.net/20.500.11821/42
[18] https://github.com/kwonmha/bert-vocab-builder
[19] https://github.com/google/sentencepiece

4   Evaluation scenarios

In this section, we describe the evaluation scenarios. First, in Section 4.1, we describe the datasets and evaluation metrics used in the evaluation tasks. In Section 4.2, we describe the settings of monolingual and cross-lingual evaluation experiments.

4.1   Datasets and evaluation metrics

We used six categories of datasets: NER, POS-tagging, dependency parsing, analogies, CoSimLex, and SuperGLUE. Each category contains datasets from several languages, and some contain several types of tasks (e.g., SuperGLUE). The categories are shortly described below.

4.1.1   Named entity recognition

In the NER experiments, we use datasets in nine languages: Croatian, English, Estonian, Finnish, Latvian, Lithuanian, Russian, Slovene and Swedish. The number of sentences and tags present in the datasets is shown in Table 1. The label sets used in datasets for different languages vary, meaning that some contain more fine-grained labels than others. To make results across different languages consistent, we trim labels in all datasets to the four common ones: location (LOC), organisation (ORG), person (PER), and "no entity" (OTHR). The latter includes every token that is not classified as any of the previous three classes. As this covers a wide variety of tokens (including named entities that do not belong to one of the three aforementioned classes, non-named entities, verbs, stopwords, etc.), we ignore the OTHR label during the evaluation. That is, we only take into account the classification scores of LOC, ORG, and PER classes.

Table 1: The collected datasets for NER task and their properties: the number of sentences and tagged words.

Language	Dataset	Sentences	Tags
Croatian	hr500k	24794	28902
English	CoNLL-2003 NER	20744	43979
Estonian	Estonian NER corpus	14287	20965
Finnish	FiNER data	14484	16833
Latvian	LV Tagger train data	9903	11599
Lithuanian	TildeNER	5500	7000
Russian	factRuEval-2016	4907	9666
Slovene[20]	ssj500k	9489	9440
Swedish	Swedish NER	9369	7292

---

[20]The Slovene ssj500k originally contains more sentences, but only 9489 are annotated with named entities.

4.1.2 POS-tagging and Dependency parsing

We used datasets in nine languages (Croatian, English, Estonian, Finnish, Latvian, Lithuanian, Russian, Slovene and Swedish) to test models on the POS-tagging and DP tasks. The datasets are obtained from the Universal Dependencies 2.3 [Nivre et al., 2018], except the Lithuanian ALKSNIS dataset, which comes from the Universal Dependencies 2.8. The number of sentences and tokens is shown in Table 2. We used 17 Universal POS tags for the POS-tagging task as they are the same in all languages and did not predict language-specific XPOS tags.

Table 2: POS-tagging and dependency parsing datasets and their properties: the treebank, number of sentences, number of tokens, and information about the size of the splits.

Language	Treebank	Tokens	Sentences	Train	Validation	Test
Croatian	SET	197044	8889	6983	849	1057
English	EWT	254854	16622	12543	2002	2077
Estonian	EDT	434245	30723	24384	3125	3214
Finnish	TDT	202208	15136	12217	1364	1555
Latvian	LVTB	152706	9920	7163	1304	1453
Lithuanian	ALKSNIS	70051	3642	2341	617	684
Russian	GSD	99389	5030	3850	579	601
Slovene	SSJ	140670	8000	6478	734	788
Swedish	Talbanken	96858	6026	4303	504	1219

We use two evaluation metrics in the dependency parsing task, the mean of unlabeled and labelled attachment scores (UAS and LAS) on the test set. The UAS and LAS are standard accuracy metrics in dependency parsing. The UAS score is defined as the proportion of tokens that are assigned the correct syntactic head. The LAS score is the proportion of tokens that are assigned the correct syntactic head and the correct dependency label [Jurafsky and Martin, 2009].

4.1.3 CoSimLex

In contrast to other datasets which are used to evaluate the performance of embeddings on specific tasks, the CoSimLex task [Armendariz et al., 2020] allows direct investigation of embeddings' properties. CoSimLex contains pairs of words and their similarity ratings assigned by human annotators. The crucial difference to previous such datasets is that the words appeared within a short text (context) when presented to the human annotators. Therefore, the word similarity ratings take the context into account, making the dataset suitable to evaluate the contextualised embeddings. The dataset is based on pairs of words from SimLex-999 [Hill et al., 2015] to allow comparison with the context-independent case. CoSimLex consists of 340 word-pairs in English, 112 in Croa-

tian, 111 in Slovene, and 24 in Finnish. Each pair is rated within two different contexts, giving a total of 1174 scores of contextual similarity.

As the example in Figure 1 shows, for each pair of words, two different contexts are presented in which these two words appear. The words in contexts produce two similarity scores, each related to one of the contexts, calculated as the mean of all the annotators' ratings for that context. This is accompanied by two standard deviation scores. and the four inflected forms of the words exactly as they appear in the contexts. Note that in the morphologically rich languages (such as Slovene, Croatian, and Finnish), many inflections of the two words are possible.

Figure 1: An example from the English CoSimLex, showing a word pair with two contexts, each with the mean and standard deviation of human similarity judgements. The original SimLex values for the same word pair without context are shown for comparison. The p-Value shown results from the Mann-Whitney U test for similarity of distributions, showing that the human judgements differ significantly between the two contexts.

Word1: man **Word2: warrior**	**SimLex**: μ 4.72 σ 1.03
Context1	**Context1:** μ 7.88 σ 2.07
When Jaimal died in the war, Patta Sisodia took the command, but he too died in the battle. These young **men** displayed true Rajput chivalry. Akbar was so impressed with the bravery of these two **warriors** that he commissioned a statue of Jaimal and Patta riding on elephants at the gates of the Agra fort.	
Context2	**Context2:** μ 3.27 σ 2.87
She has a dark past when her whole family was massacred, leaving her an orphan. By day, Shi Yeon is an employee at a natural history museum. By night, she's a top-ranking woman **warrior** in the Nine-Tailed Fox clan, charged with preserving the delicate balance between **man** and fox.	
	p-Value: 1.3×10^{-6}

Model performance is evaluated using two metrics, which measure different aspects of prediction quality:

M1 - Predicting Changes: The first metric measures the ability of a model to predict the *change in similarity ratings between the two contexts* for each word pair. This is evaluated via the correlation between the changes predicted by the system and those derived from human ratings, using the uncentered Pearson correlation. This gives a measure of the accuracy of predicting the relative magnitude of changes and allows for differences in scaling while maintaining the effect of the direction of change. The standard centered correlation normalises on the mean, so it could give high values even when a system predicts changes in the wrong direction, but with a similar distribution over examples.

$$M1 = CC_{uncentered} = \frac{\sum_{i=1}^{n}(x_i)(y_i)}{\sqrt{(\sum_{i=1}^{n} x_i)^2 (\sum_{i=1}^{n} y_i)^2}}$$

M2 - Predicting Ratings: The second metric measures the ability to predict the absolute similarity rating for each word pair in each context. This was evaluated using the harmonic mean of the Pearson and the Spearman correlation with gold-standard human judgements.

4.1.4   Monolingual and cross-lingual analogies

The word analogy task (x is to y as a is to b) was popularised by Mikolov et al. [2013]. The goal is to find a term y for a given term x so that the relationship between x and y best resembles the given relationship $a : b$. In the used datasets, There are two main groups of categories: semantic and syntactic. To illustrate a semantic relationship (country and its capital), consider, for example, that the word pair $a : b$ is given as "Finland : Helsinki". The task is to find the term y corresponding to the relationship "Sweden : y", with the expected answer being $y =$ Stockholm. In syntactic categories, each category refers to a grammatical feature, e.g., adjective degrees of comparison. The two words in any given pair then have a common stem (or even the same lemma); e.g., given the word pair "long : longer", we have an adjective in its base form and the same adjective in the comparative form. The task is to find the term y corresponding to the relationship "dark : y", with the expected answer being $y =$ darker, i.e. a comparative form of the adjective dark.

In the vector space, the analogy task is transformed into vector arithmetic. We search for nearest neighbours, i.e. we compute the distance d between vectors: $d(\text{vec(Finland)}, \text{vec(Helsinki)})$ and search for word y which would give the closest result in distance $d(\text{vec(Sweden)}, \text{vec}(y))$. We use the monolingual and cross-lingual analogy datasets in nine languages (Croatian, English, Estonian, Finnish, Latvian, Lithuanian, Russian, Slovenian, and Swedish) [Ulčar et al., 2020]. Here, the analogies are already prespecified so we do not search for the closest result but only check if the prespecified word is indeed the closest; alternatively, we measure the distance between the given pairs. The proportion of correctly identified words in the five nearest vectors forms a statistic called accuracy@5, which we report as the result.

In the cross-lingual setting, for two languages L_1 and L_2, the word analogy task matches each relation in one language with each relation from the same category in the other language. For cross-lingual contextual mappings, the presented word analogy task is not well-suited as it only contains words without their context. We describe our approach for applying this task in Section 4.2.4.

4.1.5   SuperGLUE tasks

SuperGLUE (Super General Language Understanding Evaluation) [Wang et al., 2019a] is a benchmark for testing natural language understanding (NLU) of models. It is styled after the GLUE benchmark [Wang et al., 2019b], but much more challenging. It provides a single-number metric for each of its tasks that enables the comparison and progress of NLP models. The tasks are diverse and comprised of question answering (BoolQ, COPA, MultiRC, and ReCoRD

tasks), natural language inference (CB and RTE tasks), coreference resolution (WSC), and word sense disambiguation (WiC). Non-expert humans evaluated all the tasks to give a human baseline to machine systems. Please refer to the original paper for an extensive description of the tasks.

To evaluate cross-lingual transfer and test specifics of morphologically rich languages, we translated the SuperGLUE datasets to Slovene. We used partially human translation (HT) and partially machine translation (MT). The details are presented in Table 3. Some datasets are too large (BoolQ, MultiRC, ReCoRD, RTE) to be fully human translated with our budget. We thus provide ratios between the human translated and the original English sizes. For MT from English to Slovene, we used the GoogleMT Cloud service. In our evaluation, we use six of the original eight tasks.

The WSC dataset cannot be machine-translated because it requires human assistance and verification. First, GoogleMT translations cannot handle the correct placement of HTML tags indicating coreferences. The second reason is that in Slovene coreferences can also be expressed with verbs, while coreferences in English are mainly nouns, proper names and pronouns. This makes the task more difficult in Slovene compared to English because solutions cover more types of words.

We did not include ReCoRD in the Slovene benchmark due to the low quality of the resulting dataset, consisting of confusing and ambiguous examples. Besides imperfect translations, there are differences between English and Slovene ReCoRD tasks due to the morphological richness of Slovene. In Slovene, the correct declension of a query is often not present in the text, making it impossible to provide the correct answer. Finally, similarly to WSC, ReCoRD is also affected by the problem of translating HTML tags with GoogleMT.

The WiC task cannot be translated and would have to be conceived anew because it is impossible to transfer the same set of meanings of a given word from English to a target language.

4.1.6   Terminology alignment

Terms are single words or multi-word expressions denoting concepts from specific subject fields. The bilingual terminology alignment task aligns terms between two candidate term lists in two different languages. The primary purpose of terminology alignment is to build a bilingual term bank - i.e. a list of terms in one language and their equivalents in another language.

Given a pair of terms "t_1" and "t_2", where t_1 is from one language and t_2 is its equivalent from the second language, we measured the cosine distance between vector of t_1 and vectors of all terms from the second language. If the vector of t_2 is the closest to t_1 among all terms, we count the pair as correctly aligned. For example, for a pair of terms from the Slovenian-English term bank "računovodstvo - accounting", we map the Slovene word embedding of the word "računovodstvo" from Slovene to English and check among all English word vectors for the vector that is the closest to the mapped Slovenian vector for "računovodstvo". If the closest vector is "accounting", we count this as a

Table 3: The number of instances in the original English and translated Slovene SuperGLUE tasks. HT stands for human translation and MT for machine translation. The "ratio" indicates the ratio between the number of human translated instances and all instances.

Dataset	split	English	HT	ratio	MT
BoolQ	train	9427	92	0.0098	yes
	val	3270	18	0.0055	yes
	test	3245	30	0.0092	yes
CB	train	250	250	1.0000	yes
	val	56	56	1.0000	yes
	test	250	250	1.0000	yes
COPA	train	400	400	1.0000	yes
	val	100	100	1.0000	yes
	test	500	500	1.0000	yes
MultiRC	train	5100	15	0.0029	yes
	val	953	3	0.0031	yes
	test	1800	30	0.0167	yes
ReCoRD	train	101000	60	0.0006	/
	val	10000	6	0.0006	/
	test	10000	30	0.0030	/
RTE	train	2500	232	0.0928	yes
	val	278	29	0.1043	yes
	test	300	29	0.0967	yes
WiC	train	6000	/	/	/
	val	638	/	/	/
	test	1400	/	/	/
WSC	train	554	554	1.0000	/
	val	104	104	1.0000	/
	test	146	146	1.0000	/

success, else we do not. This measure is called accuracy@1 score or 1NN score: the number of successes, divided by the number of all examples, in this case, dictionary pairs. A similar measure accuracy@n checks the proportion of correct translations among n closest words. In this work, we use the term alignment task to compare different embedding models.

For building contextualised vector representations of terms, we used the Europarl corpus [Koehn, 2005, Tiedemann, 2012]. For Croatian, Europarl is not available, so we used DGT translation memory [Steinberger et al., 2012] instead. We used these corpora, composed of mostly EU legislation texts available in all EU languages, to create contextual word embeddings. For single-word terms, we represent each term as the average vector of all contextual vector representations for that word, found in the corpus. For multi-word terms, we used a two-step approach. If the term appears in the corpus, we represent each term occurrence as the average vector of the words it is composed of. We then average over all the occurrences, as with single-word terms. In case the term does not appear in

the corpus, we represent it as the average of all words it is composed of, where word vectors are averaged over all occurrences in the corpus.

To evaluate the performance of embedding-based terminology alignment, we used Eurovoc [Steinberger et al., 2002], a multilingual thesaurus with more than 10,000 terms available in all EU languages. The models were evaluated for the following pairs of languages: Croatian-Slovenian, Estonian-Finnish, Latvian-Lithuanian, and English paired with each of the following: Croatian, Estonian, Finnish, Latvian, Lithuanian, Slovenian, and Swedish. For each language pair, we evaluated the terminology alignment in both directions, i.e. we took terms from the first language and searched for the closest terms in the second language, then repeated the procedure by taking the second language terms and searched for the closest terms in the first language.

4.2 Evaluation settings

We split our evaluations into two categories: monolingual and cross-lingual. In the monolingual evaluation, we compare fastText, ELMo, monolingual BERT-like models (English, Russian, Finnish, Swedish, Slovene, Croatian, Estonian, and Latvian), trilingual BERT models (FinEst, CroSloEngual, and LitLat BERT), and massively multilingual BERT models (mBERT and XLM-R). The exact choice of compared models depends on the availability of datasets in specific languages. In the cross-lingual setting, we compare cross-lingual maps for ELMo models, massively multilingual BERT models, and trilingual BERT models. The specifics of models for individual tasks are described below.

4.2.1 Named entity recognition

For each of the compared embeddings, we tested a separate neural architecture, adapted to the specifics of the embeddings. For fastText and ELMo embeddings, we trained NER classifiers by inputting word vectors for each token in a given sentence, along with their labels. We used a model with two bidirectional LSTM layers with 2048 units. On the output, we used the time-distributed softmax layer. For ELMo embeddings, we computed a weighted average of the three embedding vectors for each token, by learning the weights during the training. We used the Adam optimizer with a learning rate of 10^{-4} and trained for 5 epochs.

For BERT models, we fine-tuned each model on the NER dataset for 3 epochs. We used the code by HuggingFace[21] for NER classification.

4.2.2 POS-tagging

For training POS-tagging classifiers with fastText or ELMo embeddings, we used the same approach and hyper-parameters as described above for NER, but a different neural network architecture. We trained models with four hidden layers, three bidirectional LSTMs and one fully connected feed-forward layer.

---

[21]https://github.com/huggingface/transformers/tree/master/examples/legacy/token-classification

The three LSTM layers have 512, 512, and 256 units, respectively. The fully connected layer has 64 neurons.

For BERT models, we fine-tuned each model for 3 epochs, using the POS classification code by HuggingFace as for NER.

4.2.3 Dependency parsing

To train dependency parsers using ELMo embeddings, we used the SuPar tool by Yu Zhang[22]. SuPar is based on the deep biaffine attention [Dozat and Manning, 2017]. We modified the SuPar tool to accept ELMo embeddings on the input; specifically, we used the concatenation of the three ELMo vectors. The modified code has been made publicly available[23]. We trained the parser for 10 epochs for each language, using separately L-ELMo and EFML embeddings.

For fine-tuning BERT models, we modified the dep2label-bert tool [Strzyz et al., 2019, Gómez-Rodríguez et al., 2020] to work with newer versions of Hugging-Face's transformers library and to support both RoBERTa and BERT-based models. We used the modified tool to fine-tune all the BERT/RoBERTa models on the dependency parsing task for 10 epochs. We used the arc-Standard algorithm in transition-based sequence labelling encoding. The modified tool is publicly available[24].

4.2.4 Analogies

The word analogy task was initially designed for static embeddings. To evaluate contextual embeddings, we have to use the words of each analogy entry in a context. Such contexts may not exist in general corpora for some categories. We used a boilerplate sentence "If the term [w1] corresponds to the term [w2], then the term [w3] corresponds to the term [w4]." Here, [w1] through [w4] represent the four words from an analogy entry. We translated the boilerplate sentence to every language where a suitable analogy dataset is available (Croatian, English, Estonian, Finnish, Latvian, Lithuanian, Russian, Slovene, Swedish) [Ulčar et al., 2020].

For ELMo models, we concentrated on evaluating cross-lingual mapping approaches. Given a cross-lingual analogy entry (i.e. the first two words in one language, and the last two words in another language), we filled the boilerplate sentence in the training language with the four analogy words (two of them being in the "wrong" language) and extracted the vectors for words "w1" and "w2". We then filled the boilerplate sentence in the testing language with the same four words and extracted the vectors for words "w3" and "w4". We evaluated the quality of the mapping by measuring the distance between vector $v(w_4)$ and vector $v(w_2) - v(w_1) + v(w_3)$.

BERT models are masked language models, so we tried to exploit that in this task. We masked the word "w2" and tried to predict it, given every other

---

[22]https://github.com/yzhangcs/parser
[23]https://github.com/EMBEDDIA/supar-elmo
[24]https://github.com/EMBEDDIA/dep2label-transformers

word. In the cross-lingual setting, the sentence after the comma and the words
"w3" and "w4" were therefore given in the source/training language, while
the sentence before the comma and word "w1" were given in target/evaluation
language. The prediction for the masked word "w2" was expected in the tar-
get/testing language, as well.

4.2.5   SuperGLUE

We fine-tuned BERT models on SuperGLUE tasks using the Jiant tool [Phang et al.,
2020]. We used a single-task learning setting for each task and fine-tuned them
for 100 epochs each, with the initial learning rate of 10^{-5}. Each model was
fine-tuned using either machine translated or human translated datasets of the
same size.

4.2.6   Terminology alignment

For ELMo embeddings, we concatenated the three ELMo vectors into one 3072-
dimensional vector for each term. For BERT models, we extracted the vectors
from outputs of the last 4 layers and concatenated them to produce a 3072-
dimensional vector for each term.

5   Results

We present two sets of results. First, in Section 5.1 we evaluate monolingual
models, followed by evaluation of cross-lingual transfer in Section 5.2.

5.1   Monolingual evaluations

The monolingual evaluation is split into six subsections according to the type
of task. We start with NER, followed by POS-tagging, dependency parsing,
CoSimLex, analogies, and SuperGLUE tasks. The results shown for classifica-
tion tasks NER, POS-tagging, and dependency parsing are the averages of five
individual evaluation runs.

5.1.1   Named entity recognition

In Table 4, we present the results of fastText non-contextual baseline, compared
with two types of contextual ELMo embeddings, ELMoForManyLangs and L-
ELMo (described in Section 3.2). L-ELMo, trained on much larger datasets, is
the best in every language except Latvian. The fastText baseline lags behind
both ELMo embeddings.

   The results of BERT models are presented in Table 5. Each of the listed
BERT models was fine-tuned on NER datasets in languages where that makes
sense: monolingual and trilingual BERT models were used in languages used
in their pretraining, and massively multilingual models (mBERT and XLM-R)
were fine-tuned for all used languages.

Table 4: The comparison of fastText non-contextual baseline with two types of ELMo embeddings, EFML and L-ELMo on the NER task. The results are given as macro F_1 scores. The best model for each language is in **bold**. There is no Lithuanian EFML model.

Language	fastText	EFML	L-ELMo
Croatian	0.570	0.733	**0.810**
English	0.807	0.879	**0.922**
Estonian	0.734	0.828	**0.895**
Finnish	0.692	0.882	**0.923**
Latvian	0.557	**0.838**	0.818
Lithuanian	0.359	N/A	**0.755**
Slovenian	0.478	0.772	**0.849**
Swedish	0.663	0.829	**0.852**

Table 5: The results of NER evaluation task for multilingual BERT (mBERT), XLM-RoBERTa (XLM-R), trilingual BERT-based (TRI) and monolingual BERT-based (MONO) models. The scores are macro average F_1 scores of the three named entity classes.

Language	mBERT	XLM-R	TRI	MONO
Croatian	0.801	0.833	**0.886**	0.881
English	0.938	0.941	**0.944**	0.943
Estonian	0.900	0.913	0.930	**0.936**
Finnish	0.934	0.932	**0.957**	0.952
Latvian	0.847	0.859	**0.863**	0.145
Lithuanian	0.833	0.802	**0.863**	-
Slovenian	0.885	0.912	0.928	**0.933**
Swedish	0.844	0.875	-	**0.887**

All BERT-like models perform similarly in English. XLM-R outperforms mBERT on all languages, except Lithuanian and Finnish. Trilingual models outperform both mBERT and XLM-R on all languages and outperform most monolingual models, except Est-RoBERTa on Estonian and SloBERTa on Slovenian. The monolingual LVBERT model performs poorly, which is an indication that this model was not trained on a large enough dataset. The best performing trilingual model in English is CroSloEngual BERT. On Estonian, Est-RoBERTa ($F_1 = 0.936$) significantly outperforms EstBERT ($F_1 = 0.870$). Again, the latter does not seem to be trained on a large enough dataset. Comparing BERT results in Table 5 with ELMo results in Table 4, we can observe clear dominance of BERT models. The extracted ELMo embedding vectors are clearly not competitive to the entire pretrained BERT models on the NER task.

5.1.2 POS-tagging

In Table 6, we present the results of fastText non-contextual baseline, compared with two types of contextual ELMo embeddings, EFML and L-ELMo. Again, L-ELMo models are the best in all languages. Some results are surprisingly low, but this is the effect of the low quality of the datasets.

Table 6: The comparison of fastText non-contextual baseline with two types of ELMo embeddings (EFML and L-ELMo) on the POS-tagging task. The results are given as micro F_1 scores. The best results for each language are in **bold**. There is no Lithuanian EFML model.

Language	fastText	EFML	L-ELMo
Croatian	0.512	0.573	**0.963**
English	0.769	0.603	**0.952**
Estonian	0.640	0.508	**0.969**
Finnish	0.506	0.389	**0.966**
Latvian	0.462	0.489	**0.940**
Lithuanian	0.303	N/A	**0.316**
Russian	0.518	0.349	**0.929**
Slovenian	0.527	0.541	**0.966**
Swedish	0.275	0.313	**0.933**

Table 7: The results of POS-tagging evaluation task for multilingual BERT (mBERT), XLM-RoBERTa (XLM-R), trilingual BERT-based (TRI) and monolingual BERT-based (MONO) models expressed with F_1 scores. The best results for each language are in **bold**.

Language	mBERT	XLM-R	TRI	MONO
Croatian	0.978	0.981	**0.982**	0.981
English	0.964	**0.972**	0.968	0.967
Estonian	0.966	0.970	0.973	**0.977**
Finnish	0.961	0.977	0.976	**0.980**
Latvian	0.946	0.960	**0.966**	0.048
Lithuanian	0.934	**0.964**	0.961	-
Russian	0.974	**0.976**	-	0.975
Slovenian	0.984	0.988	0.990	**0.991**
Swedish	0.979	0.981	-	**0.988**

The results of BERT models are presented in Table 7. Each of the listed BERT models was fine-tuned on POS datasets in languages where that makes sense: monolingual and trilingual BERT models were used in languages used in their pretraining, and massively multilingual models (mBERT and XLM-R) were fine-tuned for all used languages.

The results show that trilingual models and massively multilingual BERT models are very competitive in the POS-tagging task, differences being relatively small and language-dependent. Nevertheless, for some languages the same pattern appears as in NER: in Slovenian and Estonian monolingual models are the best again, the trilingual models outperform monolingual Croatian and English models, the monolingual Latvian BERT model, which was pretrained on insufficient amounts of data, performs poorly again. The Est-RoBERTa model outperforms EstBERT again (0.977 vs. 0.961), all trilingual models score the same on English. Comparing BERT and ELMo results in Tables 6 and Table 7, we again observe a clear dominance of BERT models.

5.1.3 Dependency parsing

In Table 8, we compare two types of contextual ELMo embeddings (EFML and L-ELMo) on the dependency parsing task. L-ELMo models outperform EFML on all languages. The difference between them is very small in English and Russian, while the largest difference occurs in Slovenian and Estonian.

Table 8: The comparison of two types of ELMo embeddings (EFML and L-ELMo) on the dependency parsing task. Results are given as UAS and LAS scores. The best results for each language are typeset in **bold**. There is no Lithuanian EFML model.

| | ELMoForManyLangs | | L-ELMo | |
Language	UAS	LAS	UAS	LAS
Croatian	88.18	79.45	**91.74**	**85.84**
English	90.28	86.29	**90.53**	**87.16**
Estonian	81.19	72.50	**89.54**	**85.45**
Finnish	88.27	83.44	**90.83**	**86.86**
Latvian	87.17	80.76	**88.85**	**82.82**
Lithuanian	-	-	**82.84**	**72.16**
Russian	89.28	83.29	**89.33**	**83.54**
Slovenian	85.55	77.73	**93.70**	**91.39**
Swedish	88.03	83.09	**89.70**	**85.07**

The results of BERT models are presented in Table 9. Each of the listed BERT models was fine-tuned on POS datasets in languages where that makes sense: monolingual and trilingual BERT models were used in languages used in their pretraining, and massively multilingual models (mBERT and XLM-R) were fine-tuned for all used languages.

The results show that the differences between monolingual, trilingual, and massively multilingual BERT models are language-dependent. CroSloEngual is again the best performing trilingual model in English. EstBERT performs well on this task, but still worse than Est-RoBERTa (77.44 and 78.64 LAS score, respectively).

Table 9: The results of the evaluation in the dependency parsing task for multilingual BERT (mBERT), XLM-RoBERTa (XLM-R), trilingual BERT-based (TRI) and monolingual BERT-based (MONO) models. The results are given as LAS scores. The best results for each language are typeset in **bold**.

Language	mBERT	XLM-R	TRI	MONO
Croatian	70.38	78.39	**82.36**	-
English	83.19	**84.94**	83.91	83.55
Estonian	56.27	68.91	75.67	**78.64**
Finnish	57.22	71.12	79.96	**83.64**
Latvian	54.61	69.26	**74.32**	56.87
Lithuanian	44.08	56.61	**61.66**	-
Russian	70.00	73.47	-	**80.90**
Slovenian	68.08	79.27	**85.38**	84.41
Swedish	74.04	80.93	-	**85.83**

Surprisingly, comparing BERT and ELMo results in Tables 8 and Table 9, shows that L-ELMo models dominate in all languages. These results indicate that BERT models shall not always be the blind choice in text classification, as ELMo might still be competitive in some tasks.

5.1.4 CoSimLex

In Table 10, we compare performance on the CoSimLex word similarity in context task for two types of ELMo models (EFML and L-ELMo) and BERT models (massively multilingual mBERT and two trilingual models: CroSloEngual BERT and FinEst BERT). The performance is expressed with two metrics: M1 measures the ability to predict the change in similarity due to a change in context, measured as uncentered Spearman correlation between the predicted and actual change of similarity scores; and M2 measures the ability to predict absolute ratings of similarity in context, measured as the harmonic mean of the Spearman and Pearson correlations between predicted and actual similarity scores. See Section 4.1.3 for details.

Among ELMo models, L-ELMo models consistently outperform EFML models, producing closer scores to humans in both metrics and all four languages. Among BERT models, the trilingual models (see Section 3.4.3) do best for most languages and metrics. The exceptions are English, in which the original monolingual BERT outperforms the trilingual models in both metrics, and the monolingual Finnish model (FinBERT), which achieves the best results for M2. We note that these best-performing models significantly outperform the standard multilingual BERT (mBERT) in all cases except M1 for Finnish, where mBERT's results are similar.

Comparing ELMo and BERT models, BERT models are more successful and predict similarities closer to human assigned scores. Interestingly, looking at the

Table 10: Comparison of different ELMo and BERT embeddings on the CoSim-Lex datasets. We compare performance via the uncentered Spearman correlation between the predicted and true change in similarity scores (M1), and the harmonic mean of the Spearman and Pearson correlations between predicted and true similarity scores (M2). Trilingual and monolingual models are based on either [a] the original BERT model, [b] the Electra approach or [c] the RoBERTa model.

Model	Metric	ELMo models		BERT Models			
		EFML	L-ELMo	mBERT	XLM-R	TRI	MONO
English	M1	0.556	**0.570**	0.713	0.545	0.719[a]	**0.729[a]**
Croatian	M1	0.520	**0.662**	0.587	0.444	**0.715[a]**	0.351[b]
Slovene	M1	0.467	**0.550**	0.603	0.440	**0.673[a]**	0.574[c]
Finnish	M1	0.403	**0.452**	0.671	0.260	**0.672[a]**	0.595[a]
English	M2	0.449	**0.510**	0.573	0.440	0.601[a]	**0.653[a]**
Croatian	M2	0.433	**0.529**	0.443	0.387	**0.642[a]**	0.391[b]
Slovene	M2	0.328	**0.516**	0.516	0.355	**0.589[a]**	0.445[c]
Finnish	M2	0.403	**0.407**	0.289	0.053	0.533[a]	**0.570[a]**

different types of BERT models, the ones that are based on the original BERT model seem to do much better than more recent variants. Multilingual mBERT does significantly better than XLM-RoBERTa, and the best model for every category is based on the original BERT formulation (English BERT, CroSloEngual, FinEst and FinBERT). The monolingual Slovene model (SloBERTa) performs poorly (it is based on the same RoBERTa variant as XLM-RoBERTa). The monolingual Croatian model (BERTić), trained using the Electra approach (see Section 3.4.2), does especially poor on this task. It seems possible that the Electra and RoBERTa approaches, due to their different pretraining objectives, produce less human-like models in terms of their embedding similarities - but further experiments are required to draw stronger conclusions.

5.1.5 Analogies

In Table 11, we present the comparison between two types of ELMo embeddings (EFML and L-ELMo) on the word analogy task. We used two distance metrics to measure the distance between the expected and the actual result. The results strongly depend on the used distance metric. Using the Euclidean distance, EFML outperforms L-ELMo on five languages, and L-ELMo wins in three languages. On these three languages (Croatian, Estonian, and Slovenian), results of EFML are significantly worse than on other languages. Using cosine distance, which is more suited for high-dimensional vector spaces, L-ELMo outperforms EFML on all languages.

The results for BERT models are presented in Table 12. Recall that the task for BERT models is different from ELMo, as described in Section 4.2.4.

Table 11: The comparison of two types of ELMo embeddings (EFML and L-ELMo) on the word analogy task. Results are reported as the macro average distance between expected and actual word vector of the word w4. Two distance metrics were used: cosine (Cos) and Euclidean (Euc). The best results (shortest distance) for each language and metric are typeset in **bold**. There is no Lithuanian EFML model.

	EFML		L-ELMo	
Language	Cos	Euc	Cos	Euc
Croatian	0.652	73.54	**0.428**	**33.48**
English	0.442	**23.89**	**0.432**	42.99
Estonian	0.599	101.90	**0.435**	**42.38**
Finnish	0.459	**31.04**	**0.410**	41.65
Latvian	0.494	**30.32**	**0.466**	42.75
Lithuanian	-	-	**0.389**	**29.37**
Russian	0.495	**29.47**	**0.429**	44.24
Slovenian	0.568	99.22	**0.408**	**28.16**
Swedish	0.496	**28.71**	**0.478**	39.71

Each of the listed BERT models was used as a masked word prediction model in languages where that makes sense: monolingual and trilingual models were used for languages used in their pretraining, and massively multilingual models were tested for all used languages. BERTić (Croatian monolingual) model was not trained as a masked language model, so we omit it here.

Table 12: The results of the word analogy task expressed as Accuracy@5 for multilingual BERT (mBERT), XLM-RoBERTa (XLM-R), trilingual BERT-based (TRI) and monolingual BERT-based (MONO) models. The best results for each language are typeset in **bold**.

Language	mBERT	XLM-R	TRI	MONO
Croatian	0.090	0.138	**0.278**	-
English	0.404	0.413	**0.439**	0.114
Estonian	0.093	0.251	0.224	**0.393**
Finnish	0.067	0.208	**0.285**	0.173
Latvian	0.026	0.118	**0.170**	0.118
Lithuanian	0.036	0.107	**0.214**	-
Russian	0.102	**0.189**	-	0.000
Slovenian	0.061	0.146	0.195	**0.409**
Swedish	0.052	0.097	-	**0.239**

The results show that trilingual BERT models are strongly dominating in most languages where they exist. The exceptions are Slovenian and Estonian, where the monolingual models perform best. However, in most languages mono-

lingual models perform poorly. The latter holds also for EstBERT, which scores 0.165 accuracy@5, placing it behind every model, except mBERT. FinEst BERT is the best performing trilingual model on English for this task.

5.1.6 SuperGLUE tasks

The SuperGLUE benchmark is extensively used to compare large pretrained models in English[25]. In contrast to that, we concentrate on the Slovene translation of the SuperGLUE tasks, described in Section 4.1.5. Experiments in English have shown that ELMo embeddings are not competitive to pretrained transformer models like BERT in GLUE benchmarks [Wang et al., 2019a]. For this reason, we skip ELMo models and compare four BERT models in our experiments: monolingual Slovene SloBERTa, trilingual CroSloEngual BERT, massively multilingual mBERT (bert-base-multilingual-cased[26]) and XLM-R (xlm-roberta-base[27]). Each model was fine-tuned using either MT or HT datasets of the same size. Only the translated content varies between both translation types; otherwise, they contain exactly the same examples. The splits of instances into train, validation and test sets is the same as in the English variant (but mostly considerably smaller, see Table 3).

In our analysis, we vary the sizes of datasets, translation types, and prediction models. Table 13 shows the results together with several baselines trained on the original English datasets. Most comparisons to English baselines are unfair because the reported English models used significantly more examples (BoolQ, MultiRC) or, in the case of the BERT++ model, the English model was additionally pretrained with transfer tasks that are similar to a target one (CB, RTE, BoolQ, COPA). In terms of comparable datasets, fair comparisons are possible with the CB, COPA, and WSC.

The single-number overall average score (Avg in the second column) comprises five equally weighted tasks: BoolQ, CB, COPA, MultiRC, and RTE. In tasks with multiple metrics, we averaged those metrics to get a single task score. For the details on how the score is calculated for each task, see [Wang et al., 2019a].

Considering the Avg scores in Table 13, SloBERTa is the best performing model. On average, all BERT models, regardless of translation type, perform better than the Most Frequent baselines. From the translation type perspective, the models trained on HT datasets perform better than those trained on MT datasets by 1.5 points. The only task where MT is better than HT is MultiRC, but looking at single scores, we can observe that none of the models learned anything (all scores are below the Most Frequent baseline). There is a large gap between the Most frequent baseline and the rest of the models. Analysis of other specific tasks shows that for the BoolQ dataset all models predict the most frequent class (the testing set might be too small for reliable conclusions in BoolQ). We assume that training sample sizes are too small MultiRC and BoolQ

---

[25]https://super.gluebenchmark.com/leaderboard
[26]https://huggingface.co/bert-base-multilingual-cased
[27]https://huggingface.co/xlm-roberta-base

Table 13: The SuperGLUE benchmarks in English (upper part) and Slovene (lower part). All English results are taken from [Wang et al., 2019a]. The HT and MT labels indicate human and machine translated Slovene datasets. The best score for each task and language is in **bold**. The best average scores (Avg) for each language are <u>underlined</u>.

Task Models/Metrics	Avg	BoolQ Acc.	CB F1/Acc.	COPA Acc.	MultiRC F1$_a$/EM	ReCoRD F1/EM	RTE Acc.	WiC Acc.	WSC Acc.
Most Frequent	45.7	62.3	21.7/48.4	50.0	61.1/0.3	33.4/32.5	50.3	50.0	**65.1**
CBoW	44.7	62.1	49.0/71.2	51.6	0.0/0.4	14.0/13.6	49.7	53.0	**65.1**
BERT	69.3	77.4	75.7/83.6	70.6	70.0/24.0	**72.0/71.3**	71.6	**69.5**	64.3
BERT++	<u>73.3</u>	**79.0**	**84.7/90.4**	**73.8**	**70.0/24.1**	**72.0/71.3**	**79.0**	**69.5**	64.3
Human (est.)	89.8	89.0	95.8/98.9	100.0	81.8*/51.9*	91.7/91.3	93.6	80.0	100.0
Most Frequent (sl)	49.1	63.3	21.7/48.4	50.0	**76.4/0.6**	-	58.6	-	65.8
HT-mBERT	54.3	63.3	66.6/73.6	54.2	45.1/8.1	-	57.2	-	61.6
MT-mBERT	55.2	63.3	65.1/68.8	54.4	55.4/11.7	-	57.9	-	-
HT-CroSloEngual	55.6	63.3	62.1/72.4	58.2	53.0/8.4	-	58.6	-	56.2
MT-CroSloEngual	53.4	63.3	59.8/68.4	55.0	51.2/10.5	-	53.8	-	-
HT-SloBERTa	57.2	63.3	**74.0/76.8**	61.8	53.0/10.8	-	53.8	-	**73.3**
MT-SloBERTa	55.8	63.3	68.6/74.8	58.2	57.1/12.0	-	49.6	-	-
HT-XLM-R	53.5	63.3	66.2/73.2	50.0	53.3/0.9	-	57.2	-	65.8
MT-XLM-R	50.1	63.3	62.0/68.4	51.4	55.3/0.6	-	42.8	-	-
HT-Avg	<u>55.1</u>	**63.3**	**70.6**	**56.0**	29.1	-	**56.7**	-	**64.2**
MT-Avg	53.6	**63.3**	67.0	54.8	**31.7**	-	51.0	-	-

in these two tasks and have to be increased (we have only 92 HT examples in BoolQ and 15 HT examples in MultiRC). The same is also true for RTE.

Compared to English models, the best Slovene model (SloBERTa) achieved good results on WSC. It seems that none of the English models learned anything from WSC, but the SloBERTa model achieved a score of 73.3 (the Most Frequent baseline gives 65.8). Nevertheless, there is still a large gap compared to human performance. All models showed good performance on CB and fell somewhere between English CBoW and BERT. We expected better results on the fully human translated COPA task. We are investigating the reasons for low performance in this task. In general, SloBERTa was significantly better than the rest of models in CB, COPA, and WSC.

We conclude that the best BERT models perform well on tasks with enough training examples (CB, COPA, WSC) and show some level of language understanding above chance. Furthermore, the models benefited from human translated datasets compared to machine translation. For some datasets, we need to increase the number of training and/or testing examples. In further work, we intend to create a Slovene version of the WiC task from scratch and run experiments on the cleaned and improved ReCoRD task.

5.2 Cross-lingual evaluations

The cross-lingual evaluation is split into six subsections according to the type of task. We present results on NER, POS-tagging, dependency parsing, analogies, SuperGLUE, and terminology alignment. We train models on a source language dataset in each task and use it for classification in the target language, i.e. we test the zero-shot transfer unless specified otherwise. For NER, POS-tagging,

and dependency parsing tasks, the results are averaged over five individual evaluation runs, just like in the monolingual evaluations. Additionally, cross-lingual ELMoGAN maps were trained five times and each of the five maps was paired with one of the five classification models for each task during evaluation.

5.2.1 Named entity recognition

In Table 14, we present the results of cross-lingual transfer of contextual ELMo embeddings. We compared isomorphic mapping with Vecmap and MUSE libraries, and two non-isomorphic mappings using GANs, ELMoGAN-O (EG-O) and ELMoGAN-10k (EG-10k).

Table 14: Comparison of different methods for cross-lingual mapping of contextual ELMo embeddings evaluated on the NER task. The best Macro F_1 score for each language pair is in **bold**. The "Reference" column represents direct learning on the target language without cross-lingual transfer. The upper part of the table contains a scenario of cross-lingual transfer from English to a less-resourced language, and the lower part of the table shows a transfer between similar languages.

Source	Target	Dict.	Vecmap	EG-O	EG-10k	MUSE	Reference
English	Croatian	direct	**0.385**	0.274	0.365	0.024	0.810
English	Estonian	direct	0.554	0.693	**0.728**	0.284	0.895
English	Finnish	direct	0.672	0.705	**0.780**	0.229	0.922
English	Latvian	direct	0.499	0.644	**0.652**	0.216	0.818
English	Lithuanian	direct	0.498	0.522	**0.553**	0.208	0.755
English	Slovenian	direct	0.548	0.572	**0.676**	0.060	0.850
English	Swedish	direct	**0.786**	0.700	0.780	0.568	0.852
Croatian	Slovenian	direct	0.387	0.279	0.250	**0.418**	0.850
Croatian	Slovenian	triang	**0.731**	0.365	0.420	0.592	0.850
Estonian	Finnish	direct	**0.517**	0.339	0.316	0.278	0.922
Estonian	Finnish	triang	**0.779**	0.365	0.388	0.296	0.922
Finnish	Estonian	direct	0.477	0.305	0.324	**0.506**	0.895
Finnish	Estonian	triang	**0.581**	0.334	0.376	0.549	0.895
Latvian	Lithuanian	direct	**0.423**	0.398	0.404	0.345	0.755
Latvian	Lithuanian	triang	**0.569**	0.445	0.472	0.378	0.755
Lithuanian	Latvian	direct	0.263	0.312	0.335	**0.604**	0.818
Lithuanian	Latvian	triang	0.359	0.405	0.409	**0.710**	0.818
Slovenian	Croatian	direct	0.361	0.270	0.307	**0.485**	0.810
Slovenian	Croatian	triang	**0.566**	0.302	0.321	0.518	0.810
Average gap for the best cross-lingual transfer in each language							0.147

The upper part of the table shows a typical cross-lingual transfer learning scenario, where the model is transferred from resource-rich language (English) to less-resourced languages. In this case, the non-isomorphic ELMoGAN methods, particularly the ELMoGAN-10k variant, are superior to isomorphic mapping with Vecmap and MUSE libraries. In this scenario, ELMoGAN-10k is always the best or close to the best mapping approach. This is not always the case in

the lower part of Table 14, which shows the second most important cross-lingual transfer scenario: transfer between similar languages. In this scenario, isomorphic mappings with Vecmap and MUSE are superior. We hypothesise that the reason for the better performance of isomorphic mappings is the similarity of tested language pairs and less violation of the isomorphism assumption the Vecmap and MUSE methods make. The results of the mapping with the MUSE method support this hypothesis. While MUSE performs worst in most cases of transfer from English, the performance gap is smaller for transfer between similar languages. MUSE is sometimes the best method for similar languages, but its results fluctuate considerably between language pairs. The second possible factor explaining the results is the quality of the dictionaries, which are in general better for combinations involving English. In particular, dictionaries obtained by triangulation via English are of poor quality, and non-isomorphic translation might be more affected by imprecise anchor points.

In general, even the best cross-lingual ELMo models lag behind the reference model without cross-lingual transfer. The differences in Macro F_1 score are small for some languages (e.g., 5.5% for English-Swedish), but they are significantly larger for most languages. The average gap between the best cross-lingual model in each language and the monolingual reference is 14.7% for ELMo models.

In Table 15, we present the results of cross-lingual transfer for contextual BERT models. We compared massively multilingual BERT models (mBERT and XLM-R) with trilingual BERT models (TRI).

The results show a clear advantage of trilingual models compared to massively multilingual models. The trilingual models dominate in 11 out of 12 transfers, except the transfer from English to Estonian, where XLM-R is better for 0.1%. The results also show that the transfer from a similar language is more successful than transfer from English. The average difference between the most successful transfer from English and the most successful transfer from a similar language averaged over target languages is considerable, i.e. 4.6%.

Comparing cross-lingual transfer of ELMo (in Table 14) with variants of multilingual BERT (in Table 15), the transfer with BERT is considerably more successful. This indicates that ELMo, while useful for explicit extraction of embedding vectors, is less competitive with BERT in the prediction model transfer, especially if we consider that ELMo requires additional effort for preparation of contextual mapping datasets, while BERT does not need it.

Finally, the comparison between the best cross-lingual models (in the bottom part of Table 15) and the best monolingual models (reference scores taken from Table 5) shows that with cross-lingual transfer we lose on average 5.2%. This is a very encouraging result, showing that modern cross-lingual technologies have made significant progress and can bridge the technological gap for less-resourced languages. Further, this score is for zero-shot transfer, while a few-shot transfer (with small amounts of data in a target language) might be even closer to monolingual results.

Table 15: Comparison of multilingual BERT (mBERT), XLM-RoBERTa (XLM-R) and trilingual BERT-based (TRI) models evaluated on the NER task as a zero-shot transfer mode. The best Macro F_1 score for each language pair is in **bold**. The "Best monolingual" column represents the best result for direct learning on the target language without cross-lingual transfer. The upper part of the table contains a scenario of cross-lingual transfer from English to a less-resourced language, and the lower part of the table shows a transfer between similar languages.

Source.	Target.	mBERT	XLM-R	TRI	Best monolingual
English	Croatian	0.632	0.673	**0.814**	0.886
English	Estonian	0.799	**0.833**	0.832	0.936
English	Finnish	0.780	0.840	**0.902**	0.957
English	Latvian	0.714	0.756	**0.768**	0.863
English	Lithuanian	0.672	0.656	**0.702**	0.863
English	Slovenian	0.742	0.755	**0.847**	0.933
Slovenian	Croatian	0.751	0.769	**0.841**	0.886
Finnish	Estonian	0.809	0.833	**0.869**	0.936
Estonian	Finnish	0.832	0.881	**0.911**	0.957
Lithuanian	Latvian	0.785	0.816	**0.834**	0.863
Latvian	Lithuanian	0.718	0.731	**0.776**	0.863
Croatian	Slovenian	0.844	0.882	**0.901**	0.933
Average gap for the best cross-lingual transfer in each language					0.052

5.2.2 POS-tagging

In Table 16, we present the results of cross-lingual transfer of contextual ELMo embeddings. We compared isomorphic mapping with Vecmap and MUSE libraries and two non-isomorphic mappings using GANs (ELMoGAN-O and ELMoGAN-10k), described in Section 3.2. The upper part of the table shows a cross-lingual transfer learning scenario, where the model is transferred from resource-rich language (English) to less-resourced languages, and the lower part shows the transfer from similar languages.

The isomorphic mappings with MUSE are superior in the POS tagging task, followed by Vecmap. The non-isomorphic methods are inferior in this task. However, even the best cross-lingual ELMo models lag considerably compared to the reference model without cross-lingual transfer. The average difference in Macro F_1 score is 22.3% (not taking into account Lithuanian which has a failed monolingual ELMo model).

In Table 17, we present the results of cross-lingual transfer for contextual BERT models. We compared massively multilingual BERT models (mBERT and XLM-R) with trilingual BERT models (TRI).

The results show an advantage of trilingual models in transfer from similar languages, while in the transfer from English, the massively multilingual XLM-

Table 16: Comparison of different methods for cross-lingual mapping of contextual ELMo embeddings evaluated on the POS-tagging task. The best micro F_1 score for each language pair is in **bold**. The "Reference" column represents direct learning on the target language without cross-lingual transfer. The upper part of the table contains a scenario of cross-lingual transfer from English to a less-resourced language, and the lower part of the table shows a transfer between similar languages.

Source	Target	Dict.	Vecmap	EG-O	EG-10k	MUSE	Reference
English	Croatian	direct	**0.705**	0.629	0.620	0.687	0.963
English	Estonian	direct	0.728	0.678	0.647	**0.729**	0.969
English	Finnish	direct	**0.729**	0.531	0.578	0.715	0.966
English	Latvian	direct	**0.681**	0.625	0.607	0.655	0.940
English	Lithuanian	direct	**0.700**	0.648	0.605	0.670	0.316
English	Russian	direct	0.415	0.488	0.491	**0.665**	0.929
English	Slovenian	direct	0.719	0.637	0.584	**0.723**	0.966
English	Swedish	direct	0.839	0.688	0.649	**0.848**	0.933
Croatian	Slovenian	direct	0.551	0.421	0.435	**0.683**	0.966
Croatian	Slovenian	triang	0.734	0.434	0.461	**0.833**	0.966
Estonian	Finnish	direct	0.586	0.522	0.533	**0.706**	0.966
Estonian	Finnish	triang	0.673	0.514	0.543	**0.690**	0.966
Finnish	Estonian	direct	0.619	0.596	0.590	**0.792**	0.969
Finnish	Estonian	triang	0.703	0.603	0.583	**0.837**	0.969
Latvian	Lithuanian	direct	0.620	0.668	0.630	**0.757**	0.316
Latvian	Lithuanian	triang	0.657	0.671	0.646	**0.773**	0.316
Lithuanian	Latvian	direct	0.310	0.312	0.315	**0.321**	0.940
Lithuanian	Latvian	triang	0.303	0.316	0.315	**0.319**	0.940
Slovenian	Croatian	direct	0.558	0.467	0.495	**0.662**	0.963
Slovenian	Croatian	triang	0.735	0.492	0.502	**0.784**	0.963
Avg. gap for the best transfer in each language							0.104
Avg. gap for the best transfer in each language (without Lithuanian)							0.184

R models are more successful. The transfer from a similar language is more successful than the transfer from English, the average difference being 5.7%.

Similarly to NER, the comparison of ELMo cross-lingual transfer (in Table 16) with variants of multilingual BERT (in Table 17) shows that the transfer with BERT is considerably more successful. The comparison between the best cross-lingual models (these are various BERT models in Table 17) and the best monolingual models (reference scores taken from Table 7) shows that with the cross-lingual transfer we lose on average 8.2%.

5.2.3 Dependency parsing

In Table 18, we present the results of cross-lingual transfer of contextual ELMo embeddings. We compared isomorphic mapping with Vecmap and MUSE libraries and two non-isomorphic mappings using GANs (ELMoGAN-O and ELMoGAN-10k). The upper part of the table shows a cross-lingual transfer learning scenario, where the model is transferred from resource-rich language (English) to less-resourced languages, and the lower part shows the transfer from similar languages.

Table 17: Comparison of multilingual BERT (mBERT), XLM-RoBERTa (XLM-R) and trilingual BERT-based (TRI) models evaluated on the POS-tagging task as a zero-shot knowledge transfer. The best F_1 score for each language pair is in **bold**. The upper part of the table contains a scenario of cross-lingual transfer from English to a less-resourced language, and the lower part of the table shows a transfer between similar languages.

Source.	Target.	mBERT	XLM-R	TRI	Best monolingual
English	Croatian	0.837	**0.846**	0.827	0.982
English	Estonian	0.799	0.849	**0.851**	0.977
English	Finnish	0.799	**0.857**	0.839	0.980
English	Latvian	0.756	0.828	**0.829**	0.966
English	Lithuanian	0.766	**0.833**	0.822	0.964
English	Russian	0.812	**0.842**	-	0.976
English	Slovenian	0.807	**0.834**	0.819	0.991
English	Swedish	0.908	**0.925**	-	0.981
Slovenian	Croatian	0.900	0.910	**0.921**	0.982
Finnish	Estonian	0.834	0.887	**0.898**	0.977
Estonian	Finnish	0.813	0.889	**0.890**	0.980
Lithuanian	Latvian	0.805	0.857	**0.860**	0.966
Latvian	Lithuanian	0.821	0.890	**0.900**	0.964
Croatian	Slovenian	0.895	0.919	**0.924**	0.991
Average gap for the best cross-lingual transfer in each language					0.082

The isomorphic mappings with Vecmap are superior in the dependency parsing task, followed by MUSE. Similarly to POS-tagging, the non-isomorphic methods lag. Again, the best cross-lingual ELMo models produce considerably lower scores than the reference model without cross-lingual transfer. The average difference in UAS score is 9.89%, and in LAS it is 23.79%.

In Table 19, we present the results of cross-lingual transfer for contextual BERT models. We compared massively multilingual BERT models (mBERT and XLM-R) with trilingual BERT models (TRI).

The results show an advantage of trilingual models in transfer from English and similar languages. The transfer from a similar language is more successful than the transfer from English, the average difference being 14.74%. The comparison between the best BERT cross-lingual models (from Table 19) and the best monolingual models (reference scores taken from Table 9) shows that with the cross-lingual transfer we lose on average 16.50%.

Contrary to other tasks and similarly to monolingual setting, the comparison of ELMo cross-lingual transfer (in Table 18) with variants of multilingual BERT (in Table 19) shows that the transfer with ELMo is more successful. We hypothesise that this is the result of better ELMo source language models.

Table 18: Comparison of different contextual cross-lingual mapping methods for contextual ELMo embeddings, evaluated on the dependency parsing task. Results are reported as the unlabelled attachments score (UAS) and labelled attachment score (LAS). The best results for each language and type of transfer (from English or similar language) are typeset in **bold**. The column "Direct" stands for direct learning on the target (i.e. evaluation) language without cross-lingual transfer. The languages are represented with their international language codes ISO 639-1.

Train lang.	Eval. lang.	Dict.	Vecmap UAS	LAS	EG-O UAS	LAS	EG-10k UAS	LAS	MUSE UAS	LAS	Direct UAS	LAS
en	hr	direct	**73.96**	**60.53**	68.73	50.29	66.74	40.93	71.01	54.89	91.74	85.84
en	et	direct	**62.08**	**40.62**	52.01	30.22	44.80	24.59	58.76	34.07	89.54	85.45
en	fi	direct	**64.40**	**45.32**	50.80	25.23	42.65	22.66	55.03	37.61	90.83	86.86
en	lv	direct	**77.84**	**65.97**	68.51	49.47	67.09	39.41	76.26	63.45	88.85	82.82
en	lt	direct	**63.33**	**40.56**	50.04	31.26	50.04	31.26	58.70	37.78	82.84	72.16
en	ru	direct	**72.00**	**16.62**	60.74	8.92	60.68	8.18	65.23	14.77	89.33	83.54
en	sl	direct	**79.01**	**59.84**	68.82	48.20	67.04	43.34	77.18	56.53	93.70	91.39
en	sv	direct	82.08	72.74	74.39	59.70	73.81	59.63	**82.17**	**72.78**	89.70	85.07
hr	sl	direct	**85.47**	**72.70**	51.88	31.50	53.68	33.40	83.45	69.08	93.70	91.39
hr	sl	triang	**87.70**	**76.51**	54.34	36.32	59.61	38.83	**87.70**	76.40	93.70	91.39
et	fi	direct	**79.14**	**66.09**	55.67	36.85	51.35	30.66	76.66	60.01	90.83	86.86
et	fi	triang	**80.94**	**67.35**	52.63	29.94	52.83	28.70	76.96	63.37	90.83	86.86
fi	et	direct	**75.81**	57.32	54.69	33.99	53.27	32.28	74.96	**58.14**	89.54	85.45
fi	et	triang	**79.04**	**61.86**	53.64	32.73	53.86	30.13	76.74	60.27	89.54	85.45
lv	lt	direct	**76.43**	**54.24**	64.44	37.16	64.73	35.86	75.45	53.02	82.84	72.16
lv	lt	triang	**76.26**	**53.59**	65.91	37.91	65.45	33.62	75.12	51.14	82.84	72.16
lt	lv	direct	63.27	24.53	56.43	26.93	62.51	31.84	**73.70**	**44.62**	88.85	82.82
lt	lv	triang	61.32	27.29	61.89	29.39	61.95	30.11	**72.39**	**43.15**	88.85	82.82
sl	hr	direct	**77.89**	**62.58**	47.34	29.39	52.27	32.48	72.87	55.70	91.74	85.84
sl	hr	triang	**81.32**	**67.51**	50.96	32.82	56.17	35.96	78.63	63.96	91.74	85.84
Avg. gap for the best cross-lingual transfer in each language											9.89	23.79

Table 19: Comparison of multilingual BERT (mBERT), XLM-RoBERTa (XLM-R) and trilingual BERT-based (TRI) models evaluated on the dependency parsing task as a zero-shot knowledge transfer. The best LAS score for each language pair is in **bold**. The upper part of the table contains a scenario of cross-lingual transfer from English to a less-resourced language, and the lower part of the table shows a transfer between similar languages.

Source.	Target.	mBERT	XLM-R	TRI	Best monolingual
English	Croatian	42.13	54.00	**56.04**	82.36
English	Estonian	25.12	38.01	**42.30**	78.64
English	Finnish	29.08	43.30	**46.18**	83.64
English	Latvian	23.06	38.66	**44.93**	74.32
English	Lithuanian	23.21	35.98	**40.92**	61.66
English	Russian	43.41	**48.19**	-	80.90
English	Slovenian	38.72	53.90	**58.02**	85.38
English	Swedish	60.96	**70.79**	-	80.93
Slovenian	Croatian	52.61	63.66	**67.60**	82.36
Finnish	Estonian	37.34	53.98	**63.08**	78.64
Estonian	Finnish	42.11	59.54	**67.91**	83.64
Lithuanian	Latvian	31.26	48.10	**52.33**	74.32
Latvian	Lithuanian	28.28	48.59	**54.17**	61.66
Croatian	Slovenian	52.33	67.16	**71.76**	85.38
Avg. gap for the best cross-lingual transfer in each language					16.50

5.2.4 Cross-lingual analogies

We present the results of cross-lingual transfer of contextual ELMo embeddings in Table 20. We compared isomorphic mapping with Vecmap and MUSE libraries and two non-isomorphic mappings using GANs (ELMoGAN-O and ELMoGAN-10k). The upper part of the table shows a cross-lingual transfer between English and lower-resourced language. The lower part of the table shows a cross-lingual transfer between two similar languages.

Table 20: Comparison of different contextual cross-lingual mapping methods for contextual ELMo embeddings, evaluated on the cross-lingual analogy task. Results are reported as the macro average distance between expected and actual word vector of the word w4. Two distance metrics were used: cosine (Cos) and Euclidean (Euc). The best results (shortest distance) for each language and type of transfer (from English or similar language) are typeset in **bold**. The column "Direct" stands for monolingual evaluation on the target (i.e. evaluation) language without cross-lingual transfer. The languages are represented with their international language codes ISO 639-1.

Train lang.	Eval. lang.	Dict.	Vecmap		EG-O		EG-10k		MUSE		Direct	
			Cos	Euc	Cos	Euc	Cos	Euc	Cos	Euc	Cos	Euc
en	hr	direct	**0.603**	**23.47**	0.814	40.02	0.763	42.40	**0.603**	44.54	0.428	33.48
en	et	direct	**0.578**	**27.44**	0.791	43.74	0.752	45.01	0.588	51.32	0.435	42.38
en	fi	direct	0.645	59.26	0.745	**39.21**	0.694	40.82	**0.588**	52.45	0.410	41.65
en	lv	direct	0.635	**21.46**	0.809	44.62	0.778	46.58	**0.623**	50.79	0.466	42.75
en	lt	direct	0.697	**30.39**	0.812	38.67	0.719	40.84	**0.598**	41.55	0.389	29.37
en	ru	direct	**0.573**	64.35	0.771	**41.49**	0.705	43.28	0.574	53.20	0.429	44.24
en	sl	direct	**0.613**	**32.29**	0.836	38.42	0.731	40.07	0.664	42.92	0.408	28.16
en	sv	direct	0.615	64.66	0.787	**37.35**	0.720	38.84	**0.587**	47.11	0.478	39.71
hr	sl	direct	0.690	**7.59**	0.732	41.02	0.721	41.29	**0.592**	36.37	0.408	28.16
hr	sl	triang	0.715	23.89	0.729	40.91	0.727	41.45	**0.564**	35.22	0.408	28.16
et	fi	direct	**0.545**	**11.08**	0.796	47.04	0.775	48.08	0.549	50.27	0.410	41.65
et	fi	triang	0.816	33.33	0.799	46.50	0.759	47.97	**0.527**	49.06	0.410	41.65
fi	et	direct	0.598	**11.27**	0.685	41.99	0.653	43.10	**0.551**	48.47	0.435	42.38
fi	et	triang	0.692	30.25	0.725	41.23	0.644	43.09	**0.554**	48.42	0.435	42.38
lv	lt	direct	0.587	**11.96**	0.704	39.52	0.624	41.80	**0.563**	39.99	0.389	29.37
lv	lt	triang	0.681	**19.77**	0.711	39.81	0.621	41.77	**0.570**	40.17	0.389	29.37
lt	lv	direct	0.690	**12.10**	0.814	45.38	0.758	46.86	**0.524**	43.26	0.466	42.75
lt	lv	triang	0.704	**18.18**	0.812	45.28	0.752	46.47	**0.525**	43.36	0.466	42.75
sl	hr	direct	0.591	**6.62**	0.663	38.00	0.645	38.23	**0.526**	38.17	0.428	33.48
sl	hr	triang	0.572	**20.02**	0.665	37.45	0.651	38.44	**0.501**	36.92	0.428	33.48
Average gap for the best cross-lingual transfer in each language											0.118	-20.29

The results depend largely on the metric used for evaluation. With cosine distance, the mappings with MUSE are the best in most cases. For language pairs, where the MUSE method is not the best, it is a close second. However, with Euclidean distance, Vecmap mappings perform the best in most language pairs, especially between similar languages, where they significantly outperform even monolingual results. This can be partially explained by the fact, that the Vecmap method changes both the source and target language embeddings during the mapping. For three language pairs, English-Finnish, English-Russian, and English-Swedish, Vecmap mappings do not perform well using the Euclidean distance. In those cases, ELMoGAN-O mappings perform best.

In Table 21, we present the results of contextual BERT models on the cross-lingual analogy task. We compared massively multilingual BERT models (mBERT and XLM-R) with trilingual BERT models: Croatian-Slovene-English (CSE), Finnish-Estonian-English (FinEst), and Lithuanian-Latvian-English (Lit-Lat). Recall that in the cross-lingual setting, the word analogy task tries to match each relation in one language with each relation from the same category in the other language. For cross-lingual contextual mappings, the word analogy task is less adequate, and we apply this task to words in invented contexts. The upper part of the table shows a cross-lingual scenario from the resource-rich language (English) to less-resourced languages, and the lower part shows the transfer from similar languages.

Table 21: Comparison of multilingual BERT (mBERT), XLM-RoBERTa (XLM-R) and trilingual BERT-based (TRI) models, evaluated on the word analogy task as a zero-shot knowledge transfer. The best accuracy@5 score for each language pair is in **bold**. The upper part of the table contains a scenario of cross-lingual transfer from English to a less-resourced language, and the lower part of the table shows a transfer between similar languages.

Source.	Target.	mBERT	XLM-R	TRI	Best monolingual
English	Croatian	0.025	0.015	**0.103**	0.278
English	Estonian	0.018	0.029	**0.074**	0.393
English	Finnish	0.001	0.013	**0.114**	0.285
English	Latvian	0.006	**0.036**	0.033	0.170
English	Lithuanian	0.011	0.034	**0.042**	0.214
English	Russian	0.045	**0.088**	-	0.189
English	Slovenian	0.007	0.055	**0.091**	0.409
English	Swedish	**0.065**	0.053	-	0.239
Slovenian	Croatian	0.024	0.088	**0.139**	0.278
Finnish	Estonian	0.019	0.035	**0.073**	0.393
Estonian	Finnish	0.003	0.020	**0.137**	0.285
Lithuanian	Latvian	0.005	0.016	**0.032**	0.170
Latvian	Lithuanian	0.011	0.033	**0.068**	0.214
Croatian	Slovenian	0.013	0.086	**0.178**	0.409
Average gap for the best cross-lingual transfer in each language					0.174

The results show an advantage of trilingual models in transfer from both English and similar languages (the only difference being the transfer from English to Latvian, where the XLM-R is more successful). The transfer from a similar language is mostly more successful than the transfer from English.

5.2.5 SuperGLUE tasks

In the cross-lingual scenario, we tested three models, mBERT, CroSloEngual BERT (CSE), and XLM-R, in the model transfer between English and Slovene

(both directions). For Slovene as the source language, we used the available human translated examples. To make the comparison balanced, we only used the same examples from English datasets. We tested both zero-shot transfer (no training data in the target language) and few-shot transfer. In the few-shot training, we used 10 additional examples from the target language for each task. To achieve more statistically valid results, we randomly sampled these 10 examples five times and reported averages. The fine-tuning hyperparameters are the same as in the monolingual setup.

The results are presented in Table 22. Averaged over all tasks, some models improved the Most frequent baseline. In general, they were quite unsuccessful on BoolQ, MultiRC, and WSC but showed some promising results on COPA, RTE, and especially CB. Additional training examples in the few-shot scenario brought some visible improvements. It seems that models perform better in the English-Slovene direction than vice versa. The best performing model is XLM-R, followed by CroSloEngual BERT and mBERT.

Table 22: Cross-lingual results on human translated SuperGLUE test sets. The source (s.) and target (t.) language is either Slovene (sl) or English (en). The best results for zero-shot and few-shot scenarios are in bold.

Transfer	Model	s.	t.	Avg	BoolQ acc.	CB F1/acc.	COPA Acc.	MultiRC $F1_a$/EM	RTE Acc.	WSC Acc.
Zero-shot	CSE	en	sl	49.8	56.7	43.7/60.0	54.6	48.0/6.6	58.6	50.7
		sl	en	52.6	60.0	53.8/70	**59.6**	**56.7/9.6**	48.3	58.2
	mBERT	en	sl	47.4	56.7	36.2/57.2	50.2	47.3/8.7	55.2	64.4
		sl	en	48.3	60.0	44.6/50.4	49.8	56.2/8.7	51.7	57.5
	XLM-R	en	sl	**53.8**	**63.3**	**62.9/68.4**	53.6	48.5/0.3	**62.1**	56.2
		sl	en	51.7	**63.3**	59.1/67.2	47.2	52.9/12.9	51.7	**65.8**
Few-shot	CSE	en	sl	54.4	60.0	52.4/68.6	55.0	52.8/9.72	**65.5**	54.1
		sl	en	53.0	60.0	53.8/70.0	**59.5**	56.0/12.1	49.7	58.2
	mBERT	en	sl	50.9	60.1	53.1/66.2	50.4	50.8/9.8	53.8	64.4
		sl	en	51.3	60.7	51.8/58.2	50.3	57.2/11.1	56.5	56.8
	XLM-R	en	sl	**57.0**	**63.3**	**65.8/69.8**	53.3	**76.4/0.6**	62.1	57.4
		sl	en	53.0	**63.3**	63.0/69.6	48.3	51.4/10.6	55.8	**65.8**
	Most frequent			52.4	63.3	23.0/52.7	50.0	77.3/0.3	58.6	**65.8**

The low overall performance can be explained by a low number of training examples in the source language. If we take a closer look at specific models, we can observe that XLM-R shows good results on CB in both directions and evaluation scenarios. CroSloEngual BERT achieved similarly good result on COPA, where it is the only model that overpassed the baseline.

We can conclude that for the difficult SuperGLUE benchmark, the cross-lingual transfer is challenging but not impossible. In the future, we plan to expand the current set of experiments in several directions. First, we will train English models on the full SuperGLUE datasets and transfer them to Slovene human and machine-translated datasets. Second, we will train Slovene models on the combined machine and human translated datasets and transfer them to full English datasets. We will combine Slovene and English training sets and apply the models to both languages. Finally, we will also combine training for several tasks and test transfer learning scenarios.

5.2.6 Terminology alignment

We present the results of cross-lingual terminology alignment of contextual ELMo embeddings in Table 23. We compared the same four mapping methods as in the previous tasks. For each language pair, we evaluate the terminology alignment in both directions. That is, given the terms from the first language (source), we search for the equivalent terms in the second language (target), then we repeat in the other direction.

Results show that for the terminology alignment between English and other languages, the two non-isomorphic mappings perform the best on all language pairs. With English as the target language, ELMoGAN-10k always performs the best. In cases where English is the source language, ELMoGAN-O is usually the best. For terminology alignment between similar languages, isomorphic methods outperform the non-isomorphic methods on similar languages. In most cases, MUSE is the best method. If we just look at the best dictionary and mapping direction for each language pair, MUSE is the best in each language pair not involving English.

The terminology alignment is in most cases better from English than from a similar language as the source. The exceptions are Croatian and Finnish (as the targets).

In Table 24, we present the results of contextual embeddings, extracted from multilingual and trilingual BERT models. In the same table, we also compare the BERT embeddings with the best ELMo alignments. The results show that trilingual models significantly outperform massively multilingual models. The exception is alignment between Latvian (source) and Lithuanian (target), where mBERT and LitLat-BERT perform equally. Compared to ELMo embeddings, trilingual BERT models achieve better results on alignment between similar languages. However, ELMo outperforms BERT models on most language pairs, where source terms are in English (the exceptions are Croatian and Slovenian). The mBERT model performs poorly in most cases.

6 Conclusions

We performed a large scale evaluation of monolingual and cross-lingual contextual embedding approaches on several languages with sufficient resources. We concentrated on recently most successful contextual embeddings, in particular ELMo and BERT models. For ELMo models, we compared monolingual embeddings from two sources and several cross-lingual mappings with and without assumption of isomorphism. For BERT models, we compared monolingual models, massively multilingual models, and trilingual models. In the evaluation, we used several tasks: NER, POS-tagging, dependency parsing, CoSimLex, analogies, terminology alignment, and the SuperGLUE benchmarks.

Overall, the results show that L-ELMo models are superior to other ELMo models, but in general, there is a clear advantage of BERT models over ELMo models. In the monolingual setting, monolingual and trilingual BERT models

Table 23: Comparison of contextual cross-lingual mapping methods for ELMo embeddings, evaluated on the terminology alignment task. Results are reported as accuracy@1, based on the cosine distance metric. The best results for each language and type of transfer (from English in the upper part or from a similar language in the lower part) are typeset in **bold**. The languages are represented with their international language codes ISO 639-1. Direction in the third column represents the direction of vector mapping: from→to.

Source lang.	Target lang.	Dictionary (direction)	Vecmap	EG-O	EG-10k	MUSE
en	sl	direct (sl→en)	0.079	**0.152**	0.151	0.096
sl	en	direct (sl→en)	0.099	0.139	**0.195**	0.126
en	hr	direct (hr→en)	0.080	**0.153**	0.135	0.116
hr	en	direct (hr→en)	0.084	0.139	**0.153**	0.102
en	et	direct (et→en)	0.092	**0.177**	0.167	0.128
et	en	direct (et→en)	0.091	0.117	**0.133**	0.118
en	fi	direct (fi→en)	0.092	0.166	**0.176**	0.132
fi	en	direct (fi→en)	0.087	0.083	**0.116**	0.112
en	lv	direct (lv→en)	0.084	**0.157**	0.147	0.102
lv	en	direct (lv→en)	0.091	0.122	**0.140**	0.111
en	lt	direct (lt→en)	0.095	**0.181**	0.172	0.114
lt	en	direct (lt→en)	0.097	0.132	**0.171**	0.102
en	sv	direct (sv→en)	0.125	0.183	**0.187**	0.161
sv	en	direct (sv→en)	0.112	0.111	**0.167**	0.109
sl	hr	direct (hr→sl)	**0.109**	0.037	0.031	0.102
sl	hr	triang (hr→sl)	0.130	0.056	0.046	**0.156**
sl	hr	direct (sl→hr)	**0.109**	0.039	0.038	0.100
sl	hr	triang (sl→hr)	0.130	0.053	0.057	**0.155**
hr	sl	direct (hr→sl)	**0.084**	0.029	0.028	0.082
hr	sl	triang (hr→sl)	0.097	0.042	0.044	**0.121**
hr	sl	direct (sl→hr)	**0.084**	0.023	0.021	**0.084**
hr	sl	triang (sl→hr)	0.097	0.039	0.033	**0.121**
fi	et	direct (et→fi)	**0.130**	0.092	0.078	0.121
fi	et	triang (et→fi)	**0.130**	0.102	0.080	0.124
fi	et	direct (fi→et)	**0.129**	0.085	0.089	0.122
fi	et	triang (fi→et)	0.130	0.090	0.094	**0.145**
et	fi	direct (et→fi)	0.143	0.091	0.094	**0.167**
et	fi	triang (et→fi)	0.148	0.095	0.103	**0.166**
et	fi	direct (fi→et)	0.143	0.108	0.092	**0.166**
et	fi	triang (fi→et)	0.148	0.118	0.097	**0.189**
lv	lt	direct (lt→lv)	0.102	0.080	0.061	**0.123**
lv	lt	triang (lt→lv)	0.119	0.090	0.076	**0.134**
lv	lt	direct (lv→lt)	0.102	0.059	0.071	**0.123**
lv	lt	triang (lv→lt)	0.119	0.065	0.077	**0.128**
lt	lv	direct (lt→lv)	0.099	0.061	0.069	**0.102**
lt	lv	triang (lt→lv)	0.112	0.064	0.076	**0.116**
lt	lv	direct (lv→lt)	0.099	0.071	0.057	**0.102**
lt	lv	triang (lv→lt)	**0.112**	0.083	0.069	0.110

are very competitive, and frequently the trilingual BERT models dominate. In the cross-lingual setting, BERT models are much more successful compared to ELMo models. The trilingual models are mostly better than the massively

Table 24: Comparison of mBERT, XLM-R, and trilingual BERT-based models (TRI), evaluated on the terminology alignment task. Results are reported as accuracy@1, based on the cosine distance metric. The best results for each language pair are typeset in **bold**. The languages are represented with their international language codes ISO 639-1. The best ELMo result for each language pair (from Table 23) is in the rightmost column. The best overall results for each language pair are underlined.

Source lang.	Target lang.	mBERT	XLM-R	TRI	best ELMo
en	hr	0.054	0.049	**0.187**	0.153
hr	en	0.029	0.073	**0.230**	0.153
en	et	0.057	0.064	**0.121**	0.177
et	en	0.069	0.086	**0.183**	0.133
en	fi	0.084	0.090	**0.146**	0.176
fi	en	0.026	0.105	**0.215**	0.116
en	lv	0.068	0.059	**0.107**	0.157
lv	en	0.016	0.088	**0.156**	0.140
en	lt	0.072	0.058	**0.099**	0.181
lt	en	0.016	0.081	**0.147**	0.171
en	sl	0.060	0.055	**0.195**	0.152
sl	en	0.103	0.098	**0.284**	0.195
en	sv	0.135	**0.151**	-	0.187
sv	en	0.063	**0.178**	-	0.167
sl	hr	0.251	0.143	**0.267**	0.156
hr	sl	0.099	0.124	**0.250**	0.121
fi	et	0.063	0.130	**0.217**	0.145
et	fi	0.150	0.145	**0.233**	0.189
lt	lv	0.177	0.128	**0.206**	0.116
lv	lt	**0.195**	0.133	**0.195**	0.134

multilingual models. There are a few exceptions to these general conclusions. The main outlier is the dependency parsing task, where the L-ELMo embeddings are better than BERT models. The results also indicate that for the training of BERT-like models, large enough datasets are a prerequisite. For example, 500 million tokens used for LVBERT is not enough and even 1.1 billion tokens used for EstBERT does not guarantee good performance.

We can conclude that cross-lingual transfer of trained prediction models is feasible with the presented approaches, especially from similar languages and using specifically designed trilingual models. For several tasks, the performance of the best cross-lingual transferred models lags behind the monolingual models by only a few percent, confirming findings described in [Robnik-Šikonja et al., 2021]. The exact performance drop depends on the task and language.

In future work, it would be worth testing other forms of cross-lingual transfer, in particular different degrees of few-shot learning. While we compare the cross-lingual transfer of models with the human translation baselines in SuperGLUE

tasks, a wider comparison using more tasks would be welcome.

Acknowledgments

References

Carlos S. Armendariz, Matthew Purver, Matej Ulčar, Senja Pollak, Nikola Ljubešić, Marko Robnik-Šikonja, Mark Granroth-Wilding, and Kristiina Vaik. CoSimLex: A resource for evaluating graded word similarity in context. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC)*, pages 5880–5888, 2020.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, and Ting Liu. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 55–64, 2018.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations, ICLR*, 2019.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data. In *Proceedings of International Conference on Learning Representation ICLR*, 2018.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019. doi: 10.18653/v1/N19-1423.

Timothy Dozat and Christopher D. Manning. Deep biaffine attention for neural dependency parsing. In *Proceedings of 5th International Conference on Learning Representations, ICLR*, 2017.

Filip Ginter, Jan Hajič, Juhani Luotolahti, Milan Straka, and Daniel Zeman. CoNLL 2017 shared task - automatically annotated raw texts and word embeddings, 2017. URL `http://hdl.handle.net/11234/1-1989`.

Carlos Gómez-Rodríguez, Michalina Strzyz, and David Vilares. A unifying theory of transition-based and sequence labeling parsing. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3776–3793, 2020. doi: 10.18653/v1/2020.coling-main.336.

Felix Hill, Roi Reichart, and Anna Korhonen. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41 (4):665–695, 2015. doi: 10.1162/COLI_a_00237.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 1997.

Mandar Joshi, Omer Levy, Daniel S. Weld, and Luke Zettlemoyer. BERT for coreference resolution: Baselines and analysis. ArXiv preprint 1908.09091, 2019.

Daniel Jurafsky and James H. Martin. *Speech and Language Processing (2nd Edition)*. Prentice-Hall, Inc., USA, 2009.

Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *The Tenth Machine Translation Summit Proceedings of Conference*, pages 79–86. International Association for Machine Translation, 2005.

Yuri Kuratov and Mikhail Arkhipov. Adaptation of deep bidirectional multilingual transformers for Russian language. ArXiv preprint arXiv:1905.07213, 2019.

Y. Li, Zhenghua Li, M. Zhang, Rui Wang, S. Li, and Luo Si. Self-attentive biaffine dependency parsing. In *Proceedings of IJCAI*, 2019.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Nikola Ljubešić and Davor Lauc. BERTić - the transformer language model for Bosnian, Croatian, Montenegrin and Serbian. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*, pages 37–42, 2021.

Martin Malmsten, Love Börjeson, and Chris Haffenden. Playing with Words at the National Library of Sweden – Making a Swedish BERT. ArXiv preprint 2007.01658, 2020.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. CamemBERT: A tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, 2020.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

Joakim Nivre, Mitchell Abrams, and Željko Agić. Universal Dependencies 2.3, 2018. URL `http://hdl.handle.net/11234/1-2895`.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. 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 2227–2237, 2018. doi: 10.18653/v1/N18-1202.

Jason Phang, Phil Yeres, Jesse Swanson, Haokun Liu, Ian F. Tenney, Phu Mon Htut, Clara Vania, Alex Wang, and Samuel R. Bowman. `jiant` 2.0: A software toolkit for research on general-purpose text understanding models. `http://jiant.info/`, 2020.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67, 2020.

Julian Risch, Anke Stoll, Marc Ziegele, and Ralf Krestel. hpiDEDIS at GermEval 2019: Offensive language identification using a german BERT model. In *Proceedings of the 15th Conference on Natural Language Processing KONVENS*, 2019.

Marko Robnik-Šikonja, Kristjan Reba, and Igor Mozetič. Cross-lingual transfer of sentiment classifiers. *Slovenščina 2.0*, 9(1):1–25, 2021. doi: https://doi.org/10.4312/slo2.0.2021.1.1-25.

Julia Rodina, Yuliya Trofimova, Andrey Kutuzov, and Ekaterina Artemova. ELMo and BERT in semantic change detection for Russian. In *The 9th International Conference on Analysis of Images, Social Networks and Texts AIST*, 2020.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866, 2020. doi: 10.1162/tacl_a_00349.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. *arXiv preprint arXiv:1902.09492*, 2019.

Tadej Škvorc, Polona Gantar, and Marko Robnik-Šikonja. MICE: Mining idioms with contextual embeddings. arXiv preprint 2008.05759, 2020.

Anders Søgaard, Ivan Vulić, Sebastian Ruder, and Manaal Faruqui. *Cross-Lingual Word Embeddings*. Morgan & Claypool Publishers, 2019.

Ralf Steinberger, Bruno Pouliquen, and Johan Hagman. Cross-lingual document similarity calculation using the multilingual thesaurus Eurovoc. *Computational Linguistics and Intelligent Text Processing*, pages 101–121, 2002.

Ralf Steinberger, Andreas Eisele, Szymon Klocek, Spyridon Pilos, and Patrick Schlüter. DGT-TM: A freely available translation memory in 22 languages. In *Proceedings of the 8th international conference on Language Resources and Evaluation LREC 2012*, 2012.

Michalina Strzyz, David Vilares, and Carlos Gómez-Rodríguez. Viable dependency parsing as sequence labeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 717–723, 2019. doi: 10.18653/v1/N19-1077.

Bruno Taillé, Vincent Guigue, and Patrick Gallinari. Contextualized embeddings in named-entity recognition: An empirical study on generalization. In Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins, editors, *Advances in Information Retrieval*, pages 383–391, 2020.

Hasan Tanvir, Claudia Kittask, and Kairit Sirts. EstBERT: A pretrained language-specific BERT for Estonian. arXiv preprint 2011.04784, 2020.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, 2019. doi: 10.18653/v1/P19-1452.

Jörg Tiedemann. Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation LREC'12*, 2012.

Matej Ulčar and Marko Robnik-Šikonja. High quality ELMo embeddings for seven less-resourced languages. In *Proceedings of the 12th Language Resources and Evaluation Conference, LREC 2020*, pages 4733–4740, 2020.

Matej Ulčar and Marko Robnik-Šikonja. Cross-lingual alignments of ELMo contextual embeddings. ArXiv preprint arXiv:2106.15986, 2021.

Matej Ulčar, Kristiina Vaik, Jessica Lindström, Milda Dailidėnaitė, and Marko Robnik-Šikonja. Multilingual culture-independent word analogy datasets. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4067–4073, 2020.

Matej Ulčar and Marko Robnik-Šikonja. FinEst BERT and CroSloEngual BERT: less is more in multilingual models. In *Proceedings of Text, Speech, and Dialogue, TSD 2020*, pages 104–111, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. Multilingual is not enough: BERT for Finnish. arXiv preprint arXiv:1912.07076, 2019.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *Advances in Neural Information Processing Systems*, 32, 2019a.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*, 2019b.

Artūrs Znotiņš and Guntis Barzdiņš. LVBERT: Transformer-based model for Latvian language understanding. In *Human Language Technologies–The Baltic Perspective: Proceedings of the Ninth International Conference Baltic HLT 2020*, volume 328, page 111, 2020.