Computer Speech and Language 95 (2026) 101852

Contents lists available at ScienceDirect

Computer Speech & Language

journal homepage: www.elsevier.com/locate/csl

Mono- and cross-lingual evaluation of representation language models on less-resourced languages

Matej Ulčar [a], Aleš Žagar [a], Carlos S. Armendariz [b], Andraž Repar [c], Senja Pollak [c], Matthew Purver [b,c], Marko Robnik-Šikonja [a,*]

[a] University of Ljubljana, Faculty of Computer and Information Science, Ljubljana, Slovenia
[b] Queen Mary University of London, Cognitive Science Research Group, London, United Kingdom
[c] Jožef Stefan Institute, Ljubljana, Slovenia

ARTICLE INFO

Keywords:
Language models
Contextual embeddings
Less-resourced languages
BERT
ELMo
Cross-lingual

ABSTRACT

The current dominance of large language models in natural language processing is based on their contextual awareness. For text classification, text representation models, such as ELMo, BERT, and BERT derivatives, are typically fine-tuned for a specific problem. Most existing work focuses on English; in contrast, we present a large-scale multilingual empirical comparison of several monolingual and multilingual ELMo and BERT models using 14 classification tasks in nine languages. The results show, that the choice of best model largely depends on the task and language used, especially in a cross-lingual setting. In monolingual settings, monolingual BERT models tend to perform the best among BERT models. Among ELMo models, the ones trained on large corpora dominate. Cross-lingual knowledge transfer is feasible on most tasks already in a zero-shot setting without losing much performance.

1. Introduction

Deep neural networks have been at the forefront of natural language processing (NLP) for over a decade. The introduction of contextual embeddings, such as those from ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019), has significantly advanced many NLP tasks, often pushing performance to levels comparable to human capabilities. However, the field has seen a rapid escalation in the size and training requirements of transformer-based models (Vaswani et al., 2017), making them accessible primarily to large corporations. For instance, while training BERT, with its 110 million parameters, is feasible on 8 modest GPUs in a few days, very large generative models such as GPT-3 (Brown et al., 2020) (175 billion parameters) and its follow up models have pushed these boundaries out of reach of most researchers.

These large-scale models are predominantly trained on English data. With the exception of Chinese, no similarly sized models have been developed for other languages. That said, BERT-sized models have emerged for a variety of other languages, offering a more accessible alternative for multilingual NLP tasks.

In this paper, we mostly focus on text classification and empirically compare several mono- and cross-lingual 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,

* Corresponding author.
E-mail addresses: matej.ulcar@protonmail.com (M. Ulčar), marko.robniksikonja@fri.uni-lj.si (M. Robnik-Šikonja).

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

https://doi.org/10.1016/j.csl.2025.101852
Received 3 September 2023; Received in revised form 31 January 2025; Accepted 3 June 2025
Available online 27 June 2025

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). 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 first systematic monolingual evaluation of ELMo and BERT text representations for a set of less-resourced languages.
2. The first systematic evaluation of cross-lingual transfer using contextual ELMo and BERT models for a set of less-resourced languages.
3. The establishment of a set of mono- and cross-lingual datasets suitable for the evaluation of contextual embeddings in less-resourced languages.

The structure of the paper is as follows. Section 2 outlines the 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 the selected languages, datasets and evaluation metrics. Section 5 contains the evaluation settings and results of the evaluations. We first separately cover the ELMo and BERT models in monolingual and cross-lingual settings, followed by their comparison. 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 survey of over 150 papers investigating the information BERT contains, modifications to its training objectives and architecture, its overparameterization, and compression.

2.1. ELMo embeddings

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., 2022). 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 analyzing 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 labeling, 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.

Other works study ELMo for individual tasks like NER (Taillé et al., 2020), dependency parsing (Li et al., 2019), or diachronic changes (Rodina et al., 2020).

Although ELMo embeddings are no longer popular, they might still be useful in certain cases due to their compact contextual vectors (ELMo uses only three layers, compared to at least 14 layers in BERT). We expand on the existing works by systematically evaluating ELMo on six tasks in eight different languages, in addition to English. Further, we directly compare ELMo and BERT on two tasks in nine languages.

2.2. BERT models

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 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 BERT models and evaluate them 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.

Larger BERT models tend to perform better than smaller models. To systematically compare monolingual and cross-lingual performance for each language, we choose to compare only similarly sized BERT models. We evaluate several models for each language (if available) and select the best performing models.

2.3. Cross-lingual knowledge transfer

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 (2020a) trained several trilingual BERT models suitable 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.

Eronen et al. (2023) found that in a cross-lingual setting, using English as the source is not always the best choice. It is best to choose a high-resource language that is the closest to the target language. Choosing the best source language is not straightforward, as what is the closest language depends on the metric used. The results also depend on the task. For example, in some tasks, English was the best source; in other tasks, German proved to be a good source language for Slavic languages, and English was a relatively bad source for Germanic languages.

Lauscher et al. (2020) show that the success of cross-lingual knowledge transfer in low-level tasks, e.g. POS tagging, depends largely on language similarity between source and target languages. On tasks requiring higher-level language understanding, the amount of target language data in the pretraining phase of language models is the key factor.

Turc et al. (2021) observe that for the tasks, where it is suitable, zero-shot knowledge transfer can be improved by machine-translating the English dataset into another higher resource language (e.g. German or Russian) prior to fine-tuning the model. This holds especially for linguistically similar languages, and languages using scripts other than the Latin alphabet. However, for distant or non-related languages using the Latin alphabet, this does not generally hold true.

Cross-lingual knowledge transfer is limited in certain tasks, such as offensive language detection. These tasks are very sensitive to language and culture-specific biases. Very similar sentences can be labeled differently in each language. Furthermore, these differences can even be observed between different culture groups speaking the same language (Nozza, 2021; Zhou et al., 2023a,b).

As culture-specific biases influence the results, we attempted to choose tasks with limited cultural influence. In contrast to the above studies, which limit the number of languages to one or two, or limit the number of tasks to a handful, our analysis includes nine languages over 14 tasks. Additionally, we compare the feasibility of knowledge transfer in massively multilingual and few-lingual models. By translating SuperGLUE tasks into Slovene, we offer the first cross-lingual evaluation of multilingual BERT models in this benchmark suite; in addition, we also compare human and machine translations of this suite. Specifically, we analyze the effect of translation quality on monolingual performance, human-translated fine-tuning dataset, machine-translated dataset, zero-shot cross-lingual transfer (fine-tuning on another language), and few-shot cross-lingual transfer.

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 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, 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 to be 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 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:

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

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

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. In our evaluation, we used 300-dimensional fastText embeddings, pretrained on Common Crawl and Wikipedia corpora (Grave et al., 2018).

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 in the model 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 layer, 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 captures word polysemy. For an explicit word representation, one can use only the top layer. Still, more frequently, one combines all layers into a vector. The representation R_k 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 a word or character embedding), and h^{LM} denotes hidden layers of the 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, however, this has been shown to be less successful than vector extraction (Peters et al., 2019).

We compare three pretrained 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, 2020b). Ravishankar et al. (2021) trained a multilingual ELMo model (mELMo) on 13 different languages simultaneously. Of the languages considered in this work, only English, Finnish, Swedish, and Russian are covered by the mELMo, so we limit our evaluation of mELMo to those four languages.

3.3. Cross-lingual maps for fastText and ELMo

The word embedding spaces of different languages are similar enough to allow mapping from one space to another, even for distant languages, such as English and Vietnamese (Mikolov et al., 2013). 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 (2022).

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 MUSE method (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 so we need a different approach.

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

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

of the two processed languages and their bilingual dictionary. The dictionary alone is not sufficient, as the words are not given in the context. Therefore, we cannot 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 (2022).

To align contextual ELMo embeddings, Ulčar and Robnik-Šikonja (2022) developed four methods that take different contexts and word meanings into account. All methods require the contextual mapping datasets, described above. Two mappings follow the existing mapping techniques used for static embeddings and 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 the two non-isomorphic mapping methods are named ELMoGAN-O and ELMoGAN-10k (Ulčar and Robnik-Šikonja, 2022).

3.4. BERT embeddings

BERT (Bidirectional Encoder Representations from Transformers) embedding (Devlin et al., 2019) generalizes 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 the LM is to predict a missing token based on its neighborhood. BERT uses the transformer architecture of neural networks (Vaswani et al., 2017) in a bidirectional sense and introduces the task of next sentence prediction. The input representation of BERT is a sequence 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, character sequences—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, it is only required to add 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 maximize 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, as multilingual BERT covers 104 languages, its subword dictionary comprises tokens from all covered languages, which might not be optimal for a particular language. 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 the 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 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 gets 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. There is no well-documented monolingual BERT-based model for Croatian. We picked an Electra-based model, which we describe below. Electra (Clark et al., 2019) is an encoder transformer, related to BERT. Although a comparison between Electra and BERT would be interesting, there are no Electra models for most of the languages.

Kuratov and Arkhipov (2019) trained a monolingual Russian BERT (RuBERT) on Russian Wikipedia and news corpus. They used multilingual BERT (mBERT) to initialize 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 from scratch on a 3.3 billion token corpus, composed of news (Yle, STT), online discussions (Suomi24) and internet crawl of Finnish websites. The online discussions contained in 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.

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

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 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 digitized newspapers and also include 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) (Ulčar and Robnik-Šikonja, 2021a) and Estonian (Est-RoBERTa) (Ulčar and Robnik-Šikonja, 2021b) 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 the 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 the SloBERTa model and 40,000 tokens for the 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 (Ljubešć 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 modeling. 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 support 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.

LitBERTa (Pranckevič and Keruotis, 2021) is a Lithuanian masked language model, based on RoBERTa-base architecture. It is an uncased model, meaning it changes all the input text into lowercase; however, all the diacritics are kept. It was trained for 5 epochs on Lithuanian Wikipedia, web-crawled texts, parliamentary corpora, and some publicly available books. The corpus, in total, contained about 591 million words.

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 (2020a) constructed trilingual models featuring two similar less-resourced 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 less-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 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, 2020a), 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 corpora for FinEst BERT contain mostly web-crawled texts, including Wikipedia in all three languages, as well as Finnish news articles from STT and Estonian news articles from Ekspress Meedia. CroSloEngual BERT was trained on general Slovene corpus (containing prose, web-crawled texts,

[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
[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

newspapers, textbooks, etc.), Croatian web-crawled texts, Croatian news articles from 24sata, and English Wikipedia. Both trilingual BERT models were trained for about 40 epochs, which is approximately the same as multilingual BERT.

LitLat BERT (Ulčar and Robnik-Šikonja, 2021b) is based on the RoBERTa architecture, which has proven more robust and better performing than BERT. RoBERTa 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 subword vocabulary for LitLat BERT contains 84,200 tokens. The model was trained for 40 epochs, with a maximum sequence length of 512 tokens on Lithuanian and English Wikipedia, Lithuanian and Latvian parts of DGT corpus, LtTenTen14 corpus, Latvian parts of CoNLL 2017 corpus, Saeima corpus, and news articles from Ekspress Meedia.

4. Datasets and evaluation metrics

In this section, we describe the datasets and evaluation metrics used in the evaluation tasks. We used six categories of datasets: NER, POS-tagging, dependency parsing, analogies, CoSimLex, and SuperGLUE. Each category contains datasets from several languages, and SuperGLUE contains several types of tasks. The tasks are selected as being objective in nature, avoiding the potential for cross-cultural differences in more subjective tasks such as offensive language detection.

Note that the range of tasks covers those that can be broadly described as *syntactic* (POS tagging, dependency parsing) and as *semantic*, with the latter including both *lower-level* tasks focused on lexical aspects of meaning (Named entity recognition, Analogies, Terminology alignment) and *higher-level* tasks requiring more global understanding of language and context (CoSimLex, SuperGLUE).

We evaluated these tasks on nine different European languages. The languages and the evaluation datasets are presented briefly below.

4.1. Languages considered

In this work we selected the nine aforementioned languages: Croatian, English, Estonian, Finnish, Latvian, Lithuanian, Russian, Slovene and Swedish. From these nine languages, seven come from the *Indo-European* and two from the *Uralic* language family. More specifically, the Indo-European languages in our study cover two *Germanic* languages, Western Germanic (English) and Northern Germanic (Swedish); while others come from the *Balto-Slavic* group (Eastern Slavic (Russian), Southern Slavic (Slovene, Croatian), and Baltic (Latvian, Lithuanian)). Finnish and Estonian, in contrast, are Uralic languages from the *Finno-Ugric* family.

Croatian and **Slovenian** are South Slavic languages belonging to the Indo-European family. Croatian is mutually intelligible with Serbian, Bosnian, and Montenegrin languages, which are sometimes commonly referred to as the Serbo-Croatian language or BCMS (Kapović, 2011). Croatian and Slovenian are both part of the South Slavic dialect continuum. Both languages feature a large number of dialects, Slovenian has eight dialectal groups, which are further divided into over 40 dialects, which have a limited degree of mutual intelligibility (Greenberg, 2008). Croatian has three main groups of dialects: Chakavian, Kajkavian, and Shtokavian, which are sometimes considered different languages (e.g. Chakavian and Kajkavian have their own ISO 639-3 codes). Standard Croatian is based on Shtokavian, which is linguistically the furthest from Slovenian of the three. As a result, standard Slovenian and standard Croatian are not mutually intelligible, though their neighboring dialects might be.

Due to common history and language proximity, Croatian is understood by a large proportion of the Slovenian population (Roter, 2003).

English is a West Germanic language, belonging to the Indo-European family. A large portion of its vocabulary is of Romance origin (mostly French and Latin). Due to its position as a lingua franca, it is widely spoken as a foreign language. It has a large cultural (though not linguistic) influence on the other eight languages considered in this work. For instance, certain English idioms might be understood and used as calques in, e.g. Latvian or Croatian. This does not hold in reverse, however.

Estonian is a language that belongs to the Finnic branch of the Uralic language family. It is the official language of Estonia. It is related to Finnish, but not mutually intelligible with it. Estonian, like all Finnic languages, is an agglutinating language. However, in contrast to Finnish and other northern Finnic languages, it is more fusional and analytic (Erelt et al., 2007). **Finnish**, like Estonian, also belongs to the Finnic branch of the Uralic languages. It is an official language in Finland. Both Estonian and Finnish are completely unrelated to the other seven languages considered in this work.

Latvian and **Lithuanian** are Baltic languages belonging to the Indo-European language family. Though Latvian and Lithuanian are closely related, they differ in certain aspects, largely due to external influences. Lithuanian was more influenced by Slavic and Germanic languages, whereas Latvian had a large Finnic influence (Žilinskaitė-Šinkūnienė et al., 2019).

Russian is an East Slavic language. It is the official language of Russia and is widely understood in parts of Eastern Europe and Central Asia. In contrast to Croatian and Slovenian, which are written in the Latin alphabet, Russian is written using the Cyrillic alphabet.

Swedish is a North Germanic language, belonging to the Indo-European family. It is primarily spoken in Sweden, as well as in parts of Finland, where there is a sizeable Swedish-speaking minority. Swedish has, along with Finnish, the status of the national language of Finland, granted by the constitution.

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

Language	Sentences	Tags	Dataset
Croatian	24 794	28 902	hr500k (Ljubešić et al., 2016)
English	20 744	43 979	CoNLL-2003 NER (Tjong Kim Sang and De Meulder, 2003)
Estonian	14 287	20 965	Estonian NER corpus (Laur, 2013)
Finnish	14 484	16 833	FiNER data (Ruokolainen et al., 2020)
Latvian	9903	11 599	LV Tagger train data (Paikens et al., 2012)
Lithuanian	5500	7000	TildeNER (Pinnis, 2012)
Slovene	9489	9440	ssj500k[a] (Krek et al., 2019)
Swedish	9369	7292	Swedish NER (Klintberg, 2015)

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

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	197 044	8889	6983	849	1057
English	EWT	254 854	16 622	12 543	2002	2077
Estonian	EDT	434 245	30 723	24 384	3125	3214
Finnish	TDT	202 208	15 136	12 217	1364	1555
Latvian	LVTB	152 706	9920	7163	1304	1453
Lithuanian	ALKSNIS	70 051	3642	2341	617	684
Russian	GSD	99 389	5030	3850	579	601
Slovene	SSJ	140 670	8000	6478	734	788
Swedish	Talbanken	96 858	6026	4303	504	1219

4.2. Named entity recognition

In the NER experiments, we use datasets in eight languages: Croatian, English, Estonian, Finnish, Latvian, Lithuanian, 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), organization (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.

4.3. 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.

We use two evaluation metrics in the dependency parsing task, the mean of unlabeled and labeled 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).

4.4. CoSimLex

In contrast to other datasets that 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 human annotators. Therefore, the word similarity ratings take the context into account, making the dataset suitable for evaluating the contextual 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 Croatian, 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 Fig. 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'

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}

Fig. 1. An example from the English CoSimLex, showing a word pair with two contexts, each with the mean and standard deviation of human similarity judgments. 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 judgments differ significantly between the two contexts.

ratings for that context. This is accompanied by two standard deviation scores. Note that in morphologically rich languages (such as Slovene, Croatian, and Finnish), many inflections of the two words are possible.

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 normalizes 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. The uncentered Pearson correlation ($CC_{uncentered}$) is calculated as:

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

where x_i represents the change of similarity rating for a pair of contexts i as predicted by the system, and y_i represents the equivalent change derived from human ratings for the same pair.

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 judgments.

4.5. Monolingual and cross-lingual analogies

The word analogy task (x is to y as a is to b) was popularized 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 analogy 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 the nearest neighbors, i.e. we compute the distance d between vectors: d(vec(Finland), vec(Helsinki)) and search for word y which would give the closest result in the distance d(vec(Sweden), 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.

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	101 000	60	0.0006	/
	val	10 000	6	0.0006	/
	test	10 000	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	/

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 cross-lingually in Section 5.1.4.

4.6. SuperGLUE tasks

SuperGLUE (Super General Language Understanding Evaluation) (Wang et al., 2019a) is a benchmark suite for testing the natural language understanding (NLU) of models. It is styled after the GLUE benchmark (Wang et al., 2019b), but more challenging. The benchmark provides a single-number metric for each of its tasks to enable 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.

To evaluate cross-lingual transfer and test specifics of morphologically rich languages, we used the Slovene translation of SuperGLUE datasets, containing six out of eight original tasks. The datasets are partially human-translated (HT) and partially machine-translated (MT). The details are presented in Table 3. Some datasets were too large (BoolQ, MultiRC, ReCoRD, RTE) to be fully human-translated, and the table provides the ratios between the human-translated and the original English sizes. Below we describe some relevant details of the translation procedure.

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.

The ReCoRD dataset is not part of 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.

4.7. 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.

The cosine distance measure (d_{cos}) between vectors $\mathbf{u}$ and $\mathbf{v}$ is defined as:

$$d_{cos} = 1 - \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|}$$

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 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 contextualized vector representations of terms, we used the Europarl corpus (Koehn, 2005; Tiedemann, 2012). For Croatian, Europarl is not available, so we used the 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 the 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.

5. Evaluation settings and results

We split our evaluations into three categories: ELMo (Section 5.1), BERT (Section 5.2), and comparison between ELMo and BERT (Section 5.3). In the ELMo evaluation, we compare various ELMo models described in Section 3.2 (EFML, L-ELMo, and mELMo), against the fastText embeddings baseline. In the BERT evaluation, we compare monolingual BERT-like models (English, Russian, Finnish, Swedish, Slovene, Croatian, Estonian, Latvian, and Lithuanian), trilingual BERT models (FinEst, CroSloEngual, and LitLat BERT), and massively multilingual BERT models (mBERT and XLM-R).

For each task, we first perform a monolingual evaluation, followed by a cross-lingual evaluation. In monolingual evaluation, we train a task-specific model for each language using only the training data from that language. We then evaluate that model on test data from the same language. In a cross-lingual evaluation, we perform a zero-shot evaluation, unless otherwise noted. That is, we use the task-specific models trained during the monolingual evaluation and evaluate them on the test dataset from a different language. The results shown for classification tasks are the averages of five individual evaluation runs.

We employ different approaches for ELMo and BERT for solving the evaluation tasks. While we did optimize the parameters of the employed approaches, we cannot claim that either approaches or their parameters are the best possible, thus the results cannot be directly compared across the two categories. In Section 5.3, we evaluate two tasks, CoSimLex and terminology alignment, in the exact same manner using both ELMo and BERT embeddings.

Peters et al. (2019) compared feature extraction and fine-tuning approaches for ELMo and BERT models. They reported that on average, ELMo models perform better when used for feature extraction, while BERT models perform better when fine-tuned on end tasks. Following their findings, we employ the same strategy in our evaluation: fine-tuning for transformer models and word-vector extraction for ELMo models.

For the evaluation of fastText and ELMo, we tried several different model architectures. We performed a grid search over the training hyper-parameters, using the English datasets and English embeddings, separately for fastText, L-ELMo, and EFML. We apply the best-performing set of parameters to the other languages. For mELMo, we use the same set of parameters as for L-ELMo. This does not result in the best possible performance for each language, and better model architectures may exist that we have missed in our work. Nevertheless, the aim of our work is not to present state-of-the-art results for each individual task for every language but to focus on the performance gap between English and other less-resourced languages and explore various methods of bridging that gap. We believe the results presented in Sections 5.1–5.3 give a sufficient overview of the comparative performance of each evaluated contextual embedding approach.

Table 4
The comparison of fastText non-contextual baseline with three types of ELMo embeddings, EFML, mELMo, 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, and mELMo is only available for three of the tested languages.

Language	fastText-crf	fastText-sm	EFML	mELMo	L-ELMo
Croatian	0.700	0.683	0.748	N/A	**0.846**
English	0.911	0.920	0.911	0.925	**0.936**
Estonian	0.877	0.870	0.880	N/A	**0.919**
Finnish	0.896	0.876	0.904	0.903	**0.935**
Latvian	0.751	0.706	0.851	N/A	**0.863**
Lithuanian	0.588	0.654	N/A	N/A	**0.843**
Slovenian	0.676	0.733	0.802	N/A	**0.895**
Swedish	0.835	0.828	0.854	0.834	**0.870**

5.1. ELMo evaluation

In this section, we present the evaluation results of ELMo and fastText embeddings on NER, POS-tagging, DP, and word analogy tasks. For each task, we calculated the embeddings of the training and testing datasets and used the embeddings as the input for the task-specific models. In the cross-lingual evaluation, we compare cross-lingual mapping methods, described in Section 3.3, on L-ELMo embeddings only. For the purpose of mapping, we used bilingual dictionaries extracted from Wiktionary, using the wikt2dict[19] tool (Acs, 2014). The tool allows for direct dictionary extraction, as well as triangulation via a third language. Throughout the evaluation, we used direct dictionaries (direct) and dictionaries obtained with triangulation via English (triang) (Ulčar and Robnik-Šikonja, 2022).

We also evaluate mELMo embeddings in a cross-lingual manner, without explicit mapping.

For producing the fastText embeddings, we used the Python library fasttext 0.9.2, which can calculate embeddings also for out-of-vocabulary (OOV) words. For EFML, we used the Python library elmoformanylangs 0.0.4.post2, and for extracting other ELMo embeddings, we used the Python library allennlp 0.9.0.

5.1.1. Named entity recognition

We trained NER classifiers by inputting word vectors for each token in a given sentence, along with their labels. We used a model with four hidden layers, three bidirectional LSTM layers and one fully connected time-distributed feed-forward layer. The LSTM layers use leaky ReLU activation and have 512, 512, and 256 units, respectively. The fully connected time-distributed layer has 64 units and uses ReLU activation. For the output layer, we used a time-distributed softmax layer with four neurons. 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^{-3}, batch size of 32, and trained for 20 epochs. For fastText embeddings, we used two different models. The first one has the same architecture and training parameters as the one with the ELMo embeddings. For the second one, we used the CRF classifier head instead of the final softmax layer and trained for 20 epochs using the Adam optimizer with the learning rate of $2 \cdot 10^{-3}$ and batch size of 16. Both approaches give similar results with fastText embeddings; however, we got significantly worse results using the CRF head with ELMo embeddings, compared to softmax.

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

In Table 5, we present the cross-lingual transfer results of contextual L-ELMo embeddings which showed the best performance in the monolingual setting. We compared four mapping methods: isomorphic mapping with Vecmap and MUSE libraries, and two non-isomorphic mappings using GANs, ELMoGAN-O (EG-O) and ELMoGAN-10k (EG-10k).

The upper part of the table shows a typical cross-lingual transfer learning scenario, where the model is transferred from a resource-rich language (English) to a less-resourced language. In this case, the non-isomorphic ELMoGAN methods, particularly the ELMoGAN-10k variant, and the isomorphic mapping with Vecmap perform the best. In this scenario, ELMoGAN-10k is the best mapping approach for three languages, and Vecmap is the best mapping approach for four languages. mELMo performs solidly for Finnish but not so well for Swedish when the model has been trained in English.

The lower part of Table 5 shows the second most important cross-lingual transfer scenario: transfer between similar languages. In this scenario, isomorphic mappings with Vecmap and MUSE are superior, while non-isomorphic ELMoGAN methods perform poorly. We hypothesize 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 almost all cases of transfer from English, the performance gap is smaller for transfer between similar languages. In fact, MUSE is the best method for similar languages for most language pairs, though its results fluctuate considerably between language pairs. The second factor possibly explaining the results is the quality of the dictionaries,

[19] https://github.com/juditacs/wikt2dict

Table 5
Comparison of different methods for cross-lingual mapping of ELMo embeddings evaluated on the **NER task**. The best Macro F_1 score for each language pair is in **bold**. The "Reference" column is an upper bound representing 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	mELMo	Reference
English	Croatian	direct	**0.469**	0.337	0.310	0.208	–	0.846
English	Estonian	direct	0.698	0.702	**0.721**	0.513	–	0.919
English	Finnish	direct	0.751	0.635	**0.756**	0.490	0.716	0.935
English	Latvian	direct	**0.633**	0.620	0.620	0.464	–	0.863
English	Lithuanian	direct	**0.562**	0.548	0.555	0.496	–	0.843
English	Slovenian	direct	0.639	0.617	**0.658**	0.300	–	0.895
English	Swedish	direct	**0.788**	0.649	0.749	0.711	0.697	0.870
Croatian	Slovenian	direct	0.622	0.305	0.329	**0.642**	–	0.895
Croatian	Slovenian	triang	**0.806**	0.350	0.346	0.732	–	0.895
Estonian	Finnish	direct	**0.535**	0.365	0.347	0.389	–	0.935
Estonian	Finnish	triang	**0.775**	0.413	0.411	0.415	–	0.935
Finnish	Estonian	direct	0.521	0.326	0.330	**0.659**	–	0.919
Finnish	Estonian	triang	0.663	0.379	0.395	**0.668**	–	0.919
Latvian	Lithuanian	direct	0.345	0.373	0.377	**0.457**	–	0.843
Latvian	Lithuanian	triang	**0.592**	0.458	0.449	0.457	–	0.843
Lithuanian	Latvian	direct	0.385	0.320	0.310	**0.643**	–	0.863
Lithuanian	Latvian	triang	0.448	0.406	0.394	**0.738**	–	0.863
Slovenian	Croatian	direct	0.359	0.297	0.315	**0.528**	–	0.846
Slovenian	Croatian	triang	0.600	0.326	0.319	**0.616**	–	0.846
Average gaps for the best cross-lingual transfer:								
– in each language								0.162
– from English (excl. Swedish)								0.250
– from a similar language (excl. Swedish)								0.184

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	mELMo	L-ELMo
Croatian	0.956	0.977	N/A	**0.979**
English	0.930	**0.957**	0.929	**0.957**
Estonian	0.930	0.957	N/A	**0.974**
Finnish	0.922	0.966	0.822	**0.974**
Latvian	0.919	0.956	N/A	**0.958**
Lithuanian	0.902	N/A	N/A	**0.957**
Russian	0.922	**0.966**	0.934	**0.966**
Slovenian	0.954	0.977	N/A	**0.987**
Swedish	0.939	0.976	0.933	**0.981**

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 large for most languages, though they can be significantly smaller than the average for some languages (e.g., 8.2% for English–Swedish). The average gap between the best cross-lingual model in each language and the monolingual reference is 16.2% for ELMo models.

5.1.2. POS-tagging

For training POS-tagging classifiers with fastText or ELMo embeddings, we used the same approach and neural network architecture as for the NER task. The only difference is the classification head, which has 17 neurons, not 4, due to the number of possible class values. We trained the models for 20 epochs using the Adam optimizer with a learning rate of $2 \cdot 10^{-3}$ for fastText and EFML, and $5 \cdot 10^{-3}$ for other ELMo models, and a batch size of 32.

In Table 6, we present the results of the fastText non-contextual baseline, compared with three types of contextual ELMo embeddings: EFML, mELMo, and L-ELMo. Again, L-ELMo models are the best in all languages. The fastText embeddings are a strong baseline in this task, outperforming mELMo model in Finnish and Swedish. EFML models perform on par with L-ELMo on English and Russian. The gap in other languages is also relatively small.

In Table 7, we present the results of the 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.

Table 7
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 shows a transfer between similar languages.

Source	Target	Dict.	Vecmap	EG-O	EG-10k	MUSE	mELMo	Reference
English	Croatian	direct	**0.728**	0.602	0.581	0.561	–	0.979
English	Estonian	direct	**0.734**	0.658	0.643	0.650	–	0.974
English	Finnish	direct	**0.668**	0.566	0.575	0.599	0.650	0.974
English	Latvian	direct	**0.664**	0.584	0.549	0.552	–	0.958
English	Lithuanian	direct	**0.683**	0.629	0.599	0.582	–	0.957
English	Russian	direct	**0.624**	0.526	0.491	0.600	0.618	0.966
English	Slovenian	direct	**0.718**	0.572	0.537	0.480	–	0.987
English	Swedish	direct	**0.790**	0.679	0.629	0.746	0.748	0.981
Croatian	Slovenian	direct	**0.697**	0.347	0.365	0.513	–	0.987
Croatian	Slovenian	triang	**0.813**	0.373	0.393	0.646	–	0.987
Estonian	Finnish	direct	**0.654**	0.438	0.442	0.488	–	0.974
Estonian	Finnish	triang	**0.715**	0.428	0.454	0.510	–	0.974
Finnish	Estonian	direct	**0.697**	0.453	0.439	0.478	–	0.974
Finnish	Estonian	triang	**0.744**	0.448	0.459	0.488	–	0.974
Latvian	Lithuanian	direct	**0.701**	0.522	0.513	0.534	–	0.957
Latvian	Lithuanian	triang	**0.737**	0.541	0.523	0.548	–	0.957
Lithuanian	Latvian	direct	**0.633**	0.371	0.429	0.404	–	0.958
Lithuanian	Latvian	triang	**0.595**	0.411	0.393	0.419	–	0.958
Slovenian	Croatian	direct	**0.695**	0.349	0.372	0.494	–	0.979
Slovenian	Croatian	triang	**0.810**	0.384	0.421	0.559	–	0.979
Average gaps for the best cross-lingual transfer:								
– in each language								0.235
– from English (excl. Russian and Swedish)								0.272
– from a similar language (excl. Russian and Swedish)								0.230

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

	ELMoForManyLangs		L-ELMo		mELMo	
Language	UAS	LAS	UAS	LAS	UAS	LAS
Croatian	88.18	79.45	**91.74**	**85.84**	–	–
English	90.28	86.29	**90.53**	**87.16**	90.31	86.72
Estonian	81.19	72.50	**89.54**	**85.45**	–	–
Finnish	88.27	83.44	**90.83**	**86.86**	86.24	80.87
Latvian	87.17	80.76	**88.85**	**82.82**	–	–
Lithuanian	N/A	N/A	**82.84**	**72.16**	–	–
Russian	89.28	83.29	**89.33**	**83.54**	88.57	83.07
Slovenian	85.55	77.73	**93.70**	**91.39**	–	–
Swedish	88.03	83.09	**89.70**	**85.07**	85.80	79.42

The isomorphic mappings with Vecmap are superior in the POS tagging task. The non-isomorphic methods and MUSE method are inferior in this task and perform roughly on par with each other. In the cross-lingual evaluation, mELMo embeddings lag a bit behind Vecmap but outperform the other methods. However, even the best cross-lingual ELMo models lag considerably compared to the reference model without cross-lingual transfer. The average difference in micro F_1 score is 23.5%.

5.1.3. *Dependency parsing*

To train dependency parsers using ELMo embeddings, we used the SuPar tool version 1.0.0 by Yu Zhang.[20] 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. We made the modified code publicly available.[21] We trained the parser for 10 epochs for each language, using separately L-ELMo, mELMo, and EFML embeddings.

In Table 8, we compare three types of contextual ELMo embeddings (EFML, mELMo, and L-ELMo) on the DP 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, similarly to the POS-tagging task. Like in the NER and POS-tagging tasks, mELMo achieves competitive performance on English and Russian but performs worse in Finnish and Swedish.

[20] https://github.com/yzhangcs/parser
[21] https://github.com/EMBEDDIA/supar-elmo

Table 9
Comparison of different contextual cross-lingual mapping methods for contextual ELMo embeddings, evaluated on the **dependency parsing** task. Results are reported on gold segmentation as the unlabeled attachments score (UAS) and labeled 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	Eval.		Vecmap		EG-O		EG-10k		MUSE		Direct	
lang.	lang.	Dict.	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS	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
Average gaps for the best cross-lingual transfer:												
– in each language											9.89	23.79
– from English (excluding Russian and Swedish)											14.48	31.95
– from a similar language (excluding Russian and Swedish)											9.73	22.07

In Table 9, we present the results of the 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.

The isomorphic mappings with Vecmap are superior in the DP 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%. The mELMo embeddings perform worse than mappings with Vecmap, according to the UAS, but are comparable or better to other mappings. Based on the LAS, mELMo embeddings perform poorly, except for Russian, where they outperform all mapping methods. The UAS scores for the English-trained mELMo model are 56.21 (evaluated on Finnish), 60.92 (on Russian), and 75.65 (on Swedish). The LAS scores are 16.72 (on Finnish), 19.38 (on Russian), and 52.72 (on Swedish).

5.1.4. 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. To avoid this difficulty, we used the generic sentence "If the term $[w_1]$ corresponds to the term $[w_2]$, then the term $[w_3]$ corresponds to the term $[w_4]$." Here, $[w_1]$ through $[w_4]$ represent the four words from an analogy entry. We translated the generic 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 generic sentence in the training language with the four analogy words (two of them being in the second language) and extracted the vectors for words w_1 and w_2. We then filled the generic sentence in the testing language with the same four words and extracted the vectors for words w_3 and w_4. We evaluated the quality of the mapping by measuring the distance between vector $v(w_4)$ and vector $v(w_2) - v(w_1) + v(w_3)$. Each of the extracted vectors $v(w_1), v(w_2), v(w_3)$, and $v(w_4)$ is obtained by concatenating the three ELMo embeddings of that word.

In Table 10, we present the comparison between three types of ELMo embeddings (EFML, L-ELMo, and mELMo) 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 performs the best on five languages, and L-ELMo wins in three languages. On these three languages (Croatian, Estonian, and Slovenian), the results of EFML are significantly worse than on other languages. Using cosine distance, which is better suited for high-dimensional vector spaces, L-ELMo outperforms EFML and mELMo on all languages.

Table 10
The comparison of three types of ELMo embeddings (EFML, L-ELMo, and mELMo) on the **word analogy** task. Results are reported as the macro average distance between the expected and actual word vector of the word w_4. 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		mELMo	
Language	Cos	Euc	Cos	Euc	Cos	Euc
Croatian	0.652	73.54	**0.428**	**33.48**	–	–
English	0.442	**23.89**	**0.432**	42.99	0.459	37.81
Estonian	0.599	101.90	**0.435**	**42.38**	–	–
Finnish	0.459	**31.04**	**0.410**	41.65	0.475	34.51
Latvian	0.494	**30.32**	**0.466**	42.75	–	–
Lithuanian	–	–	**0.389**	**29.37**	–	–
Russian	0.495	**29.47**	**0.429**	44.24	0.460	34.05
Slovenian	0.568	99.22	**0.408**	**28.16**	–	–
Swedish	0.496	**28.71**	**0.478**	39.71	0.487	36.58

Table 11
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 the expected and actual word vector of the word w_4. 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	Eval.		Vecmap		EG-O		EG-10k		MUSE		mELMo		Direct	
lang.	lang.	Dict.	Cos	Euc	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	**0.554**	**38.82**	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	**0.557**	**39.18**	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	**0.534**	39.93	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 gaps for the best cross-lingual transfer:														
– in each language											0.118	−20.29		
– from English (excluding Russian and Swedish)											0.178	−7.26		
– from a similar language (excluding Russian and Swedish)											0.116	−26.20		

We present the results of cross-lingual transfer of contextual ELMo embeddings in Table 11. 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 less-resourced languages. The lower part of the table shows a cross-lingual transfer between two similar languages.

Multilingual mELMo performs particularly well in this task, outperforming other approaches in all the languages it supports (transfer from English to Finnish, Swedish, and Russian), except English–Swedish using the Euclidean distance. On other language pairs, the results again 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 the 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 perform the worst using the Euclidean distance. In those cases, ELMoGAN-O mappings perform the best on the English–Swedish pair, and mELMo performs the best on the English–Russian and English–Finnish pairs.

Table 12
The results of **NER task** for multilingual BERT (mBERT), XLM-RoBERTa (XLM-R), trilingual BERT-based (TRI) and monolingual BERT-based (MONO) models. The scores are macro averaged 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	0.901	0.907	0.925	**0.928**
Finnish	0.934	0.932	**0.957**	0.952
Latvian	0.849	0.867	**0.875**	0.780
Lithuanian	0.809	0.793	**0.847**	0.630
Slovenian	0.885	0.912	0.928	**0.933**
Swedish	0.844	0.875	–	**0.887**

5.2. BERT evaluation

In this section, we present the evaluation results of BERT-like models on NER, POS-tagging, DP, and word analogy tasks, as well as on the SuperGLUE suite of tasks. For each task, except for the word analogy task, we fine-tuned the entire models. The details of the evaluation procedure are presented for each task separately.

5.2.1. Named entity recognition

For BERT-like models, we added a linear classification head on top of each model, then fine-tuned the model on the NER dataset for 3 epochs with a batch size of 8, using the maximum sequence length of 512 tokens. We used the transformers library version 4.19.0, and the token classification code by HuggingFace[22] which uses the AdamW optimizer with the initial learning rate of $5 \cdot 10^{-5}$.

The results of monolingual fine-tuning of the BERT models are presented in Table 12. Multilingual models mBERT and XLM-R were fine-tuned for each language separately. The results for separate monolingual models are grouped in the same column, the same is true for trilingual models; for Croatian and Slovenian, we used CroSloEngual BERT; for Estonian and Finnish, FinEst BERT was applied; and for Latvian and Lithuanian, we used LitLat BERT. For English, we separately fine-tuned each of the three trilingual models; we report the results of the best-performing model only.

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 pretrained on a large enough dataset. The best-performing trilingual model in English is CroSloEngual BERT. On Estonian, Est-RoBERTa ($F_1 = 0.928$) significantly outperforms EstBERT ($F_1 = 0.870$). The latter also does not seem to be pretrained on a large enough dataset.

In Table 13, we present the results of zero-shot 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 in 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 the 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. 4.4%.

Comparing the cross-lingual transfer of ELMo (in Table 5) with variants of multilingual BERT (in Table 13), 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 cross-lingual knowledge transfer, especially if we consider that ELMo requires an additional effort for the 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 13) and the best monolingual models (reference scores taken from Table 12) shows that with zero-shot cross-lingual transfer we lose on average 4.9%. 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. A few-shot transfer (with small amounts of data in a target language) might be even closer to monolingual results.

5.2.2. POS-tagging

We fine-tuned each model for 3 epochs, using almost the same approach as for the NER task, described above. The only difference is that we set the maximum sequence length to 256 tokens and adjusted the classification head to classify into 17 classes instead of 4.

The results of BERT models in monolingual fine-tuning are presented in Table 14. Multilingual models mBERT and XLM-R were fine-tuned for each language separately. The results for separate monolingual models are grouped in the same column, the same is

[22] https://github.com/huggingface/transformers/tree/v4.19.0/examples/legacy/token-classification

Table 13
Comparison of multilingual BERT (mBERT), XLM-RoBERTa (XLM-R) and trilingual BERT-based (TRI) models evaluated on the **NER task** in 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	0.928
English	Finnish	0.780	0.840	**0.902**	0.957
English	Latvian	0.714	0.756	**0.768**	0.875
English	Lithuanian	0.672	0.656	**0.702**	0.847
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**	0.928
Estonian	Finnish	0.832	0.881	**0.911**	0.957
Lithuanian	Latvian	0.785	0.816	**0.834**	0.875
Latvian	Lithuanian	0.718	0.731	**0.776**	0.847
Croatian	Slovenian	0.844	0.882	**0.901**	0.933
Average gaps for the best cross-lingual transfer:					
– in each language					0.093
– from English					0.093
– from a similar language					0.049

Table 14
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 Micro F_1 scores. The best results for each language are in **bold**.

Language	mBERT	XLM-R	TRI	MONO
Croatian	0.982	0.984	**0.985**	0.984
English	0.969	**0.976**	0.973	0.971
Estonian	0.974	0.977	0.980	**0.982**
Finnish	0.970	0.982	0.982	**0.985**
Latvian	0.957	0.968	**0.972**	**0.972**
Lithuanian	0.948	**0.971**	0.968	0.935
Russian	0.977	0.980	–	**0.981**
Slovenian	0.987	0.990	0.992	**0.993**
Swedish	0.982	0.985	–	**0.989**

true for trilingual models: for Croatian and Slovenian we used CroSloEngual BERT, for Estonian and Finnish FinEst BERT, and for Latvian and Lithuanian LitLat BERT. For English, we separately fine-tuned each of the three trilingual models; we report the results of the best-performing model only.

The results show that trilingual models and massively multilingual BERT models are very competitive in the POS-tagging task, with differences being relatively small and language-dependent. Nevertheless, for some languages, the same pattern appears as in the NER task: in Slovenian and Estonian, monolingual models are the best again; the trilingual models outperform the monolingual Croatian model; and the Est-RoBERTa model outperforms EstBERT again (0.982 vs. 0.971 Micro F_1). The differences between trilingual models on English are small, CroSloEngual BERT is slightly better than the other two, while FinEst BERT performs on par with monolingual English bert-base-cased.

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 an advantage of trilingual models in transfer from similar languages, while in the transfer from English, the massively multilingual XLM-R models are more successful. The transfer from a similar language is more successful than the transfer from English, the average difference being 4.7%.

Similarly to NER, the comparison of ELMo cross-lingual transfer (in Table 7) with variants of multilingual BERT (in Table 15) 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 15) and the best monolingual models (reference scores taken from Table 14) shows that with the cross-lingual transfer we lose on average 7.1%.

5.2.3. *Dependency parsing*

For fine-tuning BERT models, we used the SuPar tool version 1.1.4. We trained a parser using the deep biaffine attention (Dozat and Manning, 2017), using a weighted sum of the hidden states of the last four layers as the output. We fine-tuned all the BERT-based models for a maximum of 30 epochs, each time selecting the checkpoint that performed the best on the validation set.

The results of BERT models are presented in Table 16. Multilingual models mBERT and XLM-R were fine-tuned for each language separately. The results for separate monolingual models are grouped in the same column, the same is true for trilingual models: for

Table 15
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 Micro 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	0.866	**0.872**	0.854	0.985
English	Estonian	0.842	**0.883**	**0.883**	0.982
English	Finnish	0.841	**0.887**	0.874	0.985
English	Latvian	0.794	**0.854**	**0.854**	0.972
English	Lithuanian	0.804	**0.860**	0.851	0.971
English	Russian	0.842	**0.861**	–	0.981
English	Slovenian	0.836	**0.859**	0.845	0.993
English	Swedish	0.919	**0.933**	–	0.989
Slovenian	Croatian	0.915	0.923	**0.931**	0.985
Finnish	Estonian	0.872	0.913	**0.921**	0.982
Estonian	Finnish	0.853	0.912	**0.913**	0.985
Lithuanian	Latvian	0.839	0.879	**0.883**	0.972
Latvian	Lithuanian	0.854	0.909	**0.917**	0.971
Croatian	Slovenian	0.908	0.930	**0.933**	0.993
Average gaps for the best cross-lingual transfer:					
– in each language					0.071
– from English (excl. Russian and Swedish)					0.112
– from a similar language (excl. Russian and Swedish)					0.065

Table 16
The results 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 reported on gold segmentation as LAS scores. The best results for each language are typeset in **bold**.

Language	mBERT	XLM-R	TRI	MONO
Croatian	87.88	88.98	89.15	**89.23**
English	90.07	**91.46**	90.40	90.44
Estonian	85.82	88.69	88.72	**90.08**
Finnish	87.96	92.24	92.01	**93.29**
Latvian	83.16	88.16	**89.10**	84.74
Lithuanian	76.22	83.63	**84.83**	73.02
Russian	88.23	**89.17**	–	88.96
Slovenian	92.66	94.48	94.84	**95.24**
Swedish	89.73	92.44	–	**92.78**

Croatian and Slovenian we used CroSloEngual BERT, for Estonian and Finnish FinEst BERT, and for Latvian and Lithuanian LitLat BERT. For English, we separately fine-tuned each of the three trilingual models; we report the results of the best-performing model only. In this case, LitLat BERT and CroSloEngual BERT performed equally well on English.

The results show that the differences between monolingual, trilingual, and massively multilingual BERT models are language-dependent. EstBERT performs relatively better on this task than other tasks, but still worse than Est-RoBERTa (87.73 and 90.08 LAS score, respectively) and most multilingual models. XLM-R outperforms monolingual models on English, Russian, Latvian, and Lithuanian. Though trilingual LitLat BERT is the best-performing model for Latvian and Lithuanian.

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 the XLM-R model in transfer from English, except for Latvian and Lithuanian. Trilingual models offer the best knowledge transfer between similar languages. The transfer from a similar language is much more successful than the transfer from English, with the average difference between transfer from English and transfer from a similar language being 9.28%. The comparison between the best BERT cross-lingual models (from Table 17) and the best monolingual models (reference scores taken from Table 16) shows that with the cross-lingual transfer, we lose on average 18.48%, which indicates that DP task is strongly language dependent.

5.2.4. 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. The same as for ELMo, we used a boilerplate sentence "If the term [w_1] corresponds to the term [w_2], then the term [w_3] corresponds to the term [w_4]." Contrary to ELMo, BERT models are masked language models, so we tried to exploit that in this task. We masked the word w_2 and tried to predict it, given every other word. In the cross-lingual setting, the sentence after the comma and the words w_3 and w_4 were given in the source language, while the sentence before the comma and word w_1 were given in the target language. The prediction for the masked word w_2 was expected in the target language,

Table 17
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	70.02	**72.18**	69.85	89.23
English	Estonian	45.81	**53.55**	51.39	90.08
English	Finnish	49.87	**59.30**	53.97	93.29
English	Latvian	48.19	56.31	**58.47**	89.10
English	Lithuanian	44.16	49.97	**50.85**	84.83
English	Russian	65.69	**71.68**	–	89.17
English	Slovenian	70.51	**73.63**	73.13	95.24
English	Swedish	77.03	**80.78**	–	92.78
Slovenian	Croatian	73.83	76.64	**78.05**	89.23
Finnish	Estonian	59.43	66.70	**67.71**	90.08
Estonian	Finnish	60.74	69.81	**70.28**	93.29
Lithuanian	Latvian	55.35	63.00	**64.19**	89.10
Latvian	Lithuanian	53.81	61.37	**61.96**	84.83
Croatian	Slovenian	77.89	79.70	**81.45**	95.24
Average gaps for the best cross-lingual transfer:					
– in each language					18.48
– from English (excl. Russian and Swedish)					28.97
– from a similar language (excl. Russian and Swedish)					19.69

Table 18
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**	0.044
Russian	0.102	**0.189**	–	0.000
Slovenian	0.061	0.146	0.195	**0.409**
Swedish	0.052	0.097	–	**0.239**

as well. Since we do not fine-tune the models for this task, we cannot talk about training and test languages in the cross-lingual evaluation. For this reason, we use the terms source language and target language, which are effectively equivalent to training language and test language used in other tasks.

The results for BERT models are presented in Table 18. Each of the listed BERT models was used as a masked word prediction model for languages that they support, as in the previous tasks. BERTić (Croatian monolingual) model was not trained as a masked language model, so we omit it here.

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. Surprisingly, in most languages, monolingual 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.

In Table 19, 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 (LitLat). 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 between similar languages.

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 the most successful). The transfer from a similar language is mostly more successful than the transfer from English.

Table 19
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 gaps for the best cross-lingual transfer:					
– in each language					0.174
– from English (excl. Russian and Swedish)					0.215
– from a similar language (excl. Russian and Swedish)					0.187

5.2.5. *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.

The SuperGLUE benchmark is extensively used to compare large pretrained models in English.[23] In contrast to that, we concentrate on the Slovene translation of the SuperGLUE tasks, described in Section 4.6. 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)[24] and XLM-R (xlm-roberta-base).[25] 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 are 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 20 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 between languages are possible with the CB, COPA, and WSC datasets.

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 20, 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 for MultiRC and BoolQ 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

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

Table 20
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 underlined.

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++	73.3	**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	55.1	**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	–	–

Table 21
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**

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 other 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, the number of training and/or testing examples has to be increased.

In the cross-lingual scenario, we tested transfer between English and Slovene (both directions) of three models, mBERT, CroSloEngual BERT (CSE), and XLM-R. 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 of hyperparameters is the same as in the monolingual setup.

The results are presented in Table 21. 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.

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

Table 22
Comparison of different ELMo and BERT embeddings on the **CoSimLex** 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.

		ELMo models		BERT Models			
Model	Metric	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]

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.

5.3. ELMo and BERT comparison

We employ exactly the same evaluation approach for ELMo and BERT models on two tasks: CoSimLex and terminology alignment. Therefore, we can directly compare the performance of one versus the other in a limited scope of these two tasks. CoSimLex is evaluated in a monolingual manner only, as it is an unsupervised task, and cross-language knowledge transfer would not make sense. The terminology alignment is evaluated in a cross-lingual manner only, as the task is aligning the terms between two different languages.

5.3.1. CoSimLex

In Table 22, we compare performance on the CoSimLex word similarity in context task for two types of ELMo models (EFML and L-ELMo) and BERT models (mBERT, XLM-R, two trilingual models: CroSloEngual BERT and FinEst BERT, and monolingual models). 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 an 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 4.4 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 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 metric. 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 consistently more successful and predict similarities closer to human-assigned scores. Interestingly, looking at the 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 poorly 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.

5.3.2. 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 the outputs of the last 4 layers (each 768-dimensional) and concatenated them to produce a 3072-dimensional vector for each term.

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 Section 5.1. For each language pair, we evaluated 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 the terminology alignment between similar languages, isomorphic methods

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 (transfer from English in the upper part of the table; from a similar language in the lower part of the table) are typeset in **bold**. The languages are represented with their international language codes ISO 639-1. The 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	mELMo
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	0.083
fi	en	direct (fi→en)	0.087	0.083	**0.116**	0.112	0.077
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	0.114
sv	en	direct (sv→en)	0.112	0.111	**0.167**	0.109	0.126
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	–

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 the alignment between Latvian (source) and Lithuanian (target), where mBERT and LitLat-BERT perform comparably. Compared to ELMo embeddings, trilingual BERT models achieve better results on alignment between similar languages. However, ELMo outperforms BERT embeddings on most language pairs where the source terms are in English (the exceptions are Croatian and Slovenian). The mBERT model performs poorly in most cases.

6. Conclusions

We conducted a comprehensive evaluation of monolingual and cross-lingual contextual embedding approaches across several languages with adequate resources. Our focus was on ELMo and BERT models, which remain stable, well-understood, and competitive in various text representation tasks—particularly in less-resourced languages and environments where computationally intensive generative language models are infeasible.

For ELMo models, we compared monolingual embeddings from two sources, as well as several cross-lingual mappings, both with and without the assumption of isomorphism. For BERT models, we evaluated monolingual, massively multilingual, and trilingual

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

variants. The evaluation encompassed a range of classification tasks, including Named Entity Recognition (NER), Part-of-Speech (POS) tagging, dependency parsing, CoSimLex, analogies, terminology alignment, and the SuperGLUE benchmarks.

In this section, we summarize our key findings and outline potential directions for future research.

Firstly, and most encouragingly, **cross-lingual transfer works:** *with the right models, and a suitable choice of language pairs, performance can be close to that of monolingual 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 for the sentiment classification task described in Robnik-Šikonja et al. (2021). Indeed, based on our evaluation, tasks intended to test higher-level language understanding, such as NER and the SuperGLUE tasks, are particularly suitable for zero-shot cross-lingual transfer. On the other hand, tasks more focused on syntax and grammar, such as dependency parsing and word analogy, are less suitable for zero-shot transfer. However, the latter is much more plausible between similar languages. This is very encouraging for building models for less-resourced languages with little annotated training data, and even without requiring massive models.

However, this comes with many caveats. Unsurprisingly given recent findings in many NLP tasks with very large language models, **size matters**, with large models generally outperforming smaller ones. BERT models generally outperform ELMo. In a similar vein, **training matters**: ELMo models trained on large corpora (L-ELMo) are superior to other ELMo models. We also found that monolingual models can even perform worse than multilingual ones if they are trained on small datasets: LVBERT (500 million tokens in the training dataset) performs worse than multilingual models on some tasks; even the 1.1 billion tokens used for EstBERT do not guarantee good performance; and the short training time of the monolingual Lithuanian model LitBERTa seems to negatively affect its performance, although it is still the best model for Lithuanian on some tasks.

Confirming the findings of Lauscher et al. (2020) and Eronen et al. (2023) (see Section 2 above), **language similarity matters** too: in cross-lingual transfer, performance usually improves with greater similarity between source and target language, and transfer similar source-target pairs usually outperforms the common default of transfer from/to English. While the cross-lingual transfer works to some degree for all the tested languages, the success of transfer depends on the transfer method, size of the models, and task. Trilingual and multilingual BERT models show particularly good cross-lingual transfer performance.

More surprisingly, however, **task matters:** the best choice of model and transfer method varies strongly with the task, and this effect can be large enough to override those mentioned above: in the terminology alignment and CoSimLex tasks (quite different from the classification tasks), although trilingual BERTs often perform best, ELMo generally beats massively multilingual BERTs and is the winner in a surprising number of settings. More generally, it is clear that **multilinguality has a cost:** monolingual models do generally perform best, and trilingual ones are generally better than massively multilingual ones. Good cross-lingual knowledge transfer, therefore, seems to require a careful choice of model and transfer setting for the task in hand Fujinuma et al. (see, e.g. 2022, who suggest matching adaptation methods to the training languages and their similarity to the target).

A further attempt to generalize conclusions across tasks and languages is given below in Table 25 and Fig. 2.

Table 25 shows a summary of some of these conclusions, comparing across models and tasks in zero-shot cross-lingual transfer settings. In general, BERT models perform better than ELMo models: the best-performing model is a BERT model in all cases except one (the terminology alignment task, for transfer from English to other languages). Within these general categories, though, the best choice of model depends on the task and the language setting, but some generalizations emerge. For transfer from English, the

Table 25
A summary of results for models across tasks, showing the best-performing embeddings in zero-shot cross-lingual transfer settings. We take a "majority vote" approach: for each task, we show the embedding model that gives best performance on the highest number of source-target language pairs. We show the best performer amongst ELMo and BERT models separately, marked with 'x'; and the best overall across all models, marked '**X**'. For SuperGLUE, only BERT results are available, and we use the average performance across the individual SuperGLUE tasks. For the Analogies task, the ELMo and BERT experiments use different metrics, so we do not give an overall best; for ELMo we take cosine distance (not Euclidean) as the metric. For the dependency parsing task, we use the LAS metric.

Task	ELMo				BERT		
Transfer from English	Vecmap	EG-O	MUSE	mELMO	mBERT	XLM-R	TRI
NER	x						**X**
POS	x					**X**	
Dependency parsing	x					**X**	
Analogies				x			x
SuperGLUE	–	–	–	–		**X**	
Terminology		**X**					x
Transfer from other	Vecmap	EG-O	MUSE	mELMO	mBERT	XLM-R	TRI
NER			x				**X**
POS	x						**X**
Dependency parsing	x						**X**
Analogies			x				x
SuperGLUE	–	–	–	–			**X**
Terminology			x				**X**

Fig. 2. A summary of results across language pairs and tasks, showing the increase in error caused by zero-shot transfer. The chart shows the absolute increase in error (1-F_1 for NER and POS tasks, 1-*acc* for dependency parsing (DEP) and Analogies (Ana) tasks).

massively multilingual XLM-R seems just as good as the more specific trilingual models (performing best in 3 out of 6 tasks), while for other language transfer pairs, the trilingual approach shows a clear advantage. With ELMo, the Vecmap approach to transfer seems generally the best, but multilingually-trained MUSE does better for non-English source languages.

Fig. 2 then shows a comparison of the effectiveness of cross-lingual transfer across tasks and languages. We model the loss in zero-shot cross-lingual transfer as the absolute increase in error, characterizing error as $1 - M$ where M is the key metric for the task (Macro-F_1 for NER, Micro-F_1 for POS, LAS accuracy score for dependency parsing (DEP) and plain accuracy for Analogies (Ana). Generally, it seems that transfer between similar language pairs is more effective (resulting in lower error increases) than transfer from English, although this is only statistically significant for the NER and POS tasks. It also seems that standard tasks based on lexical sequence labeling/classification (NER and POS) show much better transfer than the more syntactically or semantically complex tasks (dependency parsing and analogies).

However, exactly how these issues will play out as models become larger and larger and include more tasks and languages in their training data is not yet clear. The very large language models now becoming available show surprising robustness to new data and ability to transfer to new tasks using zero- or few-shot methods (see e.g. Brown et al., 2020; Dong et al., 2023); whether they help solve the cross-lingual transfer problem, and make the choice of methods simpler in less-resourced settings, will be a crucial point to investigate in the coming years. However, contrary to ELMo and BERT, very large language models are all generative,

using only the decoder stack of transformer architecture, and tend to use considerably more computational resources even at the inference time. BERT models tested in this paper use only the encoder stack of the transformer architecture, which is a much more efficient representation for classification tasks. For many users and tasks, this computational efficiency may play a significant role for a considerable time.

In future work, it will therefore be important to test other forms of cross-lingual transfer, in particular different degrees of few-shot learning, and larger models. In addition, 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.

CRediT authorship contribution statement

Matej Ulčar: Writing – review & editing, Writing – original draft, Validation, Software, Resources, Methodology, Formal analysis. **Aleš Žagar:** Writing – original draft, Validation, Software, Formal analysis, Data curation. **Carlos S. Armendariz:** Writing – original draft, Validation, Software, Resources, Methodology, Data curation. **Andraž Repar:** Writing – original draft, Validation, Resources, Data curation. **Senja Pollak:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Funding acquisition, Conceptualization. **Matthew Purver:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **Marko Robnik-Šikonja:** Writing – review & editing, Writing – original draft, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The work was partially supported by the Slovenian Research and Innovation Agency (ARIS) core research programmes P6-0411 and P2-0103, and projects L2-50070, J7-3159, GC-0002, and PoVeJMo (Adaptive Natural Language Processing with Large Language Models). The project was partially funded via the Franco-Slovene bilateral project BI-FR/23-24-PROTEUS-006. Partial support was also received from the UK EPSRC under grant EP/S033564/1, and the EPSRC/AHRC Centre for Doctoral Training in Media and Arts Technology EP/L01632X/1. This paper was also supported by the EU H2020 grant No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media), and EU ERA Chair grant no. 101186647, project AI4DH. We are grateful to SLING HPC network for access to Vega GPU partition (grant no. S24O01-42). Last, our thanks go to anonymous reviewers whose constructive comments helped us to significantly improve the paper.

Data availability

Data will be made available on request.

References

Acs, J., 2014. Pivot-based multilingual dictionary building using wiktionary. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation. LREC.

Armendariz, C.S., Purver, M., Ulčar, M., Pollak, S., Ljubešić, N., Robnik-Šikonja, M., Granroth-Wilding, M., Vaik, K., 2020. CoSimLex: A resource for evaluating graded word similarity in context. In: Proceedings of the 12th Language Resources and Evaluation Conference. LREC, pp. 5880–5888.

Artetxe, M., Labaka, G., Agirre, E., 2018. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In: Thirty-Second AAAI Conference on Artificial Intelligence.

Bojanowski, P., Grave, E., Joulin, A., Mikolov, T., 2017. Enriching word vectors with subword information. Trans. Assoc. Comput. Linguist. 5, 135–146.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D., 2020. Language models are few-shot learners. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (Eds.), In: Advances in Neural Information Processing Systems, vol. 33, Curran Associates, Inc., pp. 1877–1901, URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Che, W., Liu, Y., Wang, Y., Zheng, B., Liu, T., 2018. 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. pp. 55–64.

Clark, K., Luong, M.-T., Le, Q.V., Manning, C.D., 2019. ELECTRA: Pre-training text encoders as discriminators rather than generators. In: International Conference on Learning Representations. ICLR.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., Stoyanov, V., 2019. Unsupervised cross-lingual representation learning at scale. arXiv preprint arXiv:1911.02116.

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., Jégou, H., 2018. Word translation without parallel data. In: Proceedings of International Conference on Learning Representation ICLR.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2019. 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). pp. 4171–4186. http://dx.doi.org/10.18653/v1/N19-1423.

Dong, Q., Li, L., Dai, D., Zheng, C., Wu, Z., Chang, B., Sun, X., Xu, J., Li, L., Sui, Z., 2023. A survey on in-context learning. arXiv:2301.00234.

Dozat, T., Manning, C.D., 2017. Deep biaffine attention for neural dependency parsing. In: Proceedings of 5th International Conference on Learning Representations. ICLR.

Erelt, M., Erelt, T., Keevallik, L., Laanekask, H., Pajusalu, K., Viitso, T.-R., 2007. Estonian Language. Linguistica Uralica. Supplementary Series 1.

Eronen, J., Ptaszynski, M., Masui, F., 2023. Zero-shot cross-lingual transfer language selection using linguistic similarity. Inf. Process. Manage. 60, 103250.

Fujinuma, Y., Boyd-Graber, J., Kann, K., 2022. Match the script, adapt if multilingual: Analyzing the effect of multilingual pretraining on cross-lingual transferability. In: Muresan, S., Nakov, P., Villavicencio, A. (Eds.), Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Dublin, Ireland, pp. 1500–1512. http://dx.doi.org/10.18653/v1/2022.acl-long.106, URL: https://aclanthology.org/2022.acl-long.106.

Ginter, F., Hajič, J., Luotolahti, J., Straka, M., Zeman, D., 2017. CoNLL 2017 shared task - automatically annotated raw texts and word embeddings. URL: http://hdl.handle.net/11234/1-1989.

Grave, E., Bojanowski, P., Gupta, P., Joulin, A., Mikolov, T., 2018. Learning word vectors for 157 languages. In: Proceedings of the International Conference on Language Resources and Evaluation. LREC 2018.

Greenberg, M., 2008. A short reference grammar of Slovene, LINCOM studies in Slavic linguistics, Lincom Europa. URL: https://books.google.si/books?id=wsxlAAAAMAAJ.

Hill, F., Reichart, R., Korhonen, A., 2015. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. Comput. Linguist. 41, 665–695. http://dx.doi.org/10.1162/COLI_a_00237.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 1735–1780.

Joshi, M., Levy, O., Weld, D.S., Zettlemoyer, L., 2019. BERT for coreference resolution: Baselines and analysis. ArXiv preprint 1908.09091.

Jurafsky, D., Martin, J.H., 2009. Speech and Language Processing, second ed. Prentice-Hall, Inc., USA.

Kapović, M., 2011. Language, ideology and politics in Croatia. Slavia Centralis 4, 45–56.

Klintberg, A., 2015. Training a Swedish NER-model for Stanford CoreNLP. https://medium.com/@klintcho/training-a-swedish-ner-model-for-stanford-corenlp-part-1-3e3f281a753a, (Last Accessed 22 July 2021).

Koehn, P., 2005. Europarl: A parallel corpus for statistical machine translation. In: The Tenth Machine Translation Summit Proceedings of Conference. pp. 79–86.

Krek, S., Dobrovoljc, K., Erjavec, T., Može, S., Ledinek, N., Holz, N., Zupan, K., Gantar, P., Kuzman, T., Čibej, J., Arhar Holdt, Š., Kavčič, T., Škrjanec, I., Marko, D., Jezeršek, L., Zajc, A., 2019. Training corpus ssj500k 2.2. Slovenian language resource repository CLARIN.SI.

Kuratov, Y., Arkhipov, M., 2019. Adaptation of deep bidirectional multilingual transformers for Russian language. arXiv preprint arXiv:1905.07213.

Laur, S., 2013. Nimeüksuste Korpus. Center of Estonian Language Resources.

Lauscher, A., Ravishankar, V., Vulić, I., Glavaš, G., 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual transformers. In: Webber, B., Cohn, T., He, Y., Liu, Y. (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. EMNLP, ssociation for Computational Linguistics, pp. 4483–4499. http://dx.doi.org/10.18653/v1/2020.emnlp-main.363, Online URL: https://aclanthology.org/2020.emnlp-main.363.

Li, Y., Li, Z., Zhang, M., Wang, R., Li, S., Si, L., 2019. Self-attentive biaffine dependency parsing. In: Proceedings of IJCAI.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V., 2019. RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692.

Ljubešić, N., Klubička, F., Agić, Željko, Jazbec, I.-P., 2016. New inflectional lexicons and training corpora for improved morphosyntactic annotation of Croatian and Serbian. In: Proceedings of the LREC 2016.

Ljubešć, N., Lauc, D., 2021. BERTić - the transformer language model for Bosnian, Croatian, Montenegrin and Serbian. In: Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing. pp. 37–42.

Malmsten, M., Börjeson, L., Haffenden, C., 2020. Playing with Words at the National Library of Sweden – Making a Swedish BERT. ArXiv preprint 2007.01658, arXiv:2007.01658.

Martin, L., Muller, B., Ortiz Suárez, P.J., Dupont, Y., Romary, L., de la Clergerie, É., Seddah, D., Sagot, B., 2020. CamemBERT: A tasty French language model. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7203–7219.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J., 2013. Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems. pp. 3111–3119.

Nivre, J., Abrams, M., Agić, Ž., 2018. Universal Dependencies 2.3. URL: http://hdl.handle.net/11234/1-2895.

Nozza, D., 2021. Exposing the limits of zero-shot cross-lingual hate speech detection. In: Zong, C., Xia, F., Li, W., Navigli, R. (Eds.), Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers). Association for Computational Linguistics, pp. 907–914. http://dx.doi.org/10.18653/v1/2021.acl-short.114, Online. URL: https://aclanthology.org/2021.acl-short.114.

Paikens, P., Auziņa, I., Garkaje, G., Paegle, M., 2012. Towards named entity annotation of Latvian national library corpus. Frontiers Artificial Intelligence Appl. 247, 169–175. http://dx.doi.org/10.3233/978-1-61499-133-5-169.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L., 2018. 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). pp. 2227–2237. http://dx.doi.org/10.18653/v1/N18-1202.

Peters, M.E., Ruder, S., Smith, N.A., 2019. To tune or not to tune? Adapting pretrained representations to diverse tasks. In: Proceedings of the 4th Workshop on Representation Learning for NLP. RepL4NLP-2019, pp. 7–14. http://dx.doi.org/10.18653/v1/W19-4302.

Phang, J., Yeres, P., Swanson, J., Liu, H., Tenney, I.F., Htut, P.M., Vania, C., Wang, A., Bowman, S.R., 2020. jiant 2.0: A software toolkit for research on general-purpose text understanding models. http://jiant.info/.

Pinnis, M., 2012. Latvian and Lithuanian named entity recognition with TildeNER. In: Proceedings of the 8th International Conference on Language Resources and Evaluation. LREC 2012, pp. 1258–1265.

Pranckevič, M., Keruotis, J., 2021. LitBERTa uncased model, hugging face models repository. URL: https://huggingface.co/jkeruotis/LitBERTa-uncased.

Ravishankar, V., Kutuzov, A., Øvrelid, L., Velldal, E., 2021. Multilingual ELMo and the effects of corpus sampling. In: Proceedings of the 23rd Nordic Conference on Computational Linguistics. NoDaLiDa, pp. 378–384.

Risch, J., Stoll, A., Ziegele, M., Krestel, R., 2019. hpiDEDIS at GermEval 2019: Offensive language identification using a German BERT model. In: Proceedings of the 15th Conference on Natural Language Processing KONVENS.

Robnik-Šikonja, M., Reba, K., Mozetič, I., 2021. Cross-lingual transfer of sentiment classifiers. Slovenščina 2. 0 9, 1–25. http://dx.doi.org/10.4312/slo2.0.2021.1.1-25.

Rodina, J., Trofimova, Y., Kutuzov, A., Artemova, E., 2020. ELMo and BERT in semantic change detection for Russian. In: The 9th International Conference on Analysis of Images, Social Networks and Texts. AIST.

Rogers, A., Kovaleva, O., Rumshisky, A., 2020. A primer in BERTology: What we know about how BERT works. Trans. Assoc. Comput. Linguist. 8, 842–866. http://dx.doi.org/10.1162/tacl_a_00349.

Roter, P., 2003. Language issues in the context of 'Slovenian smallness', nation-building, ethnicity and language politics in transition countries. pp. 211–242.

Ruokolainen, T., Kauppinen, P., Silfverberg, M., Lindén, K., 2020. A Finnish news corpus for named entity recognition. Lang Resour. Eval. 54, 247–272.

Schuster, T., Ram, O., Barzilay, R., Globerson, A., 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. arXiv preprint arXiv:1902.09492.

Škvorc, T., Gantar, P., Robnik-Šikonja, M., 2022. MICE: Mining idioms with contextual embeddings. Knowl.-Based Syst. 235, 107606.
Søgaard, A., Vulić, I., Ruder, S., Faruqui, M., 2019. Cross-Lingual Word Embeddings. Morgan & Claypool Publishers.
Steinberger, R., Eisele, A., Klocek, S., Pilos, S., Schlüter, P., 2012. DGT-TM: A freely available translation memory in 22 languages. In: Proceedings of the 8th International Conference on Language Resources and Evaluation LREC 2012.
Steinberger, R., Pouliquen, B., Hagman, J., 2002. Cross-lingual document similarity calculation using the multilingual thesaurus Eurovoc. Comput. Linguist. Intell. Text Process. 101–121.
Taillé, B., Guigue, V., Gallinari, P., 2020. Contextualized embeddings in named-entity recognition: An empirical study on generalization. In: Advances in Information Retrieval. pp. 383–391.
Tanvir, H., Kittask, C., Sirts, K., 2020. EstBERT: A pretrained language-specific BERT for Estonian. arXiv preprint 2011.04784.
Tenney, I., Das, D., Pavlick, E., 2019. BERT rediscovers the classical NLP pipeline. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 4593–4601. http://dx.doi.org/10.18653/v1/P19-1452.
Tiedemann, J., 2012. Parallel data, tools and interfaces in OPUS. In: Proceedings of the Eighth International Conference on Language Resources and Evaluation. LREC'12.
Tjong Kim Sang, E.F., De Meulder, F., 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In: Proceedings of CoNLL-2003. pp. 142–147.
Turc, I., Lee, K., Eisenstein, J., Chang, M.-W., Toutanova, K., 2021. Revisiting the primacy of English in zero-shot cross-lingual transfer. arXiv preprint arXiv:2106.16171.
Ulčar, M., Robnik-Šikonja, M., 2020a. FinEst BERT and CroSloEngual BERT: less is more in multilingual models. In: Proceedings of Text, Speech, and Dialogue. TSD 2020, pp. 104–111.
Ulčar, M., Robnik-Šikonja, M., 2020b. High quality ELMo embeddings for seven less-resourced languages. In: Proceedings of the 12th Language Resources and Evaluation Conference. LREC 2020, pp. 4733–4740.
Ulčar, M., Robnik-Šikonja, M., 2021a. SloBERTa: Slovene monolingual large pretrained masked language model. In: Proceedings of the 24th International Multiconference – IS2021. SiKDD, 2021.
Ulčar, M., Robnik-Šikonja, M., 2021b. Training dataset and dictionary sizes matter in BERT models: The case of Baltic languages. In: International Conference on Analysis of Images, Social Networks and Texts. AISNT 2021, pp. 162–172.
Ulčar, M., Robnik-Šikonja, M., 2022. Cross-lingual alignments of ELMo contextual embeddings. Neural Comput. Appl. 1–19.
Ulčar, M., Vaik, K., Lindström, J., Dailidenaite, M., Robnik-Šikonja, M., 2020. Multilingual culture-independent word analogy datasets. In: Proceedings of the 12th Language Resources and Evaluation Conference. pp. 4067–4073.
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. In: Advances in Neural Information Processing Systems. pp. 5998–6008.
Virtanen, A., Kanerva, J., Ilo, R., Luoma, J., Luotolahti, J., Salakoski, T., Ginter, F., Pyysalo, S., 2019. Multilingual is not enough: BERT for Finnish. arXiv preprint arXiv:1912.07076.
Wang, A., Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R., 2019a. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. Adv. Neural Inf. Process. Syst. 32.
Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R., 2019b. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In: International Conference on Learning Representations.
Zhou, L., Cabello, L., Cao, Y., Hershcovich, D., 2023a. Cross-cultural transfer learning for Chinese offensive language detection. In: Dev, S., Prabhakaran, V., Adelani, D., Hovy, D., Benotti, L. (Eds.), Proceedings of the First Workshop on Cross-Cultural Considerations in NLP. C3NLP, Association for Computational Linguistics, Dubrovnik, Croatia, pp. 8–15. http://dx.doi.org/10.18653/v1/2023.c3nlp-1.2, URL: https://aclanthology.org/2023.c3nlp-1.2.
Zhou, L., Karamolegkou, A., Chen, W., Hershcovich, D., 2023b. Cultural compass: Predicting transfer learning success in offensive language detection with cultural features. In: Bouamor, H., Pino, J., Bali, K. (Eds.), Findings of the Association for Computational Linguistics. EMNLP 2023, Association for Computational Linguistics, Singapore, pp. 12684–12702. http://dx.doi.org/10.18653/v1/2023.findings-emnlp.845, URL: https://aclanthology.org/2023.findings-emnlp.845.
Žilinskaitė-Šinkūnienė, E., Šķilters, J., Zariņa, L., Bērziņa, N., 2019. Containmentand support: similarities and variation in Lithuanian, Latvian and Estonian. Baltistica 54, 205–255.
Znotiņš, A., Barzdiņš, G., 2020. LVBERT: Transformer-based model for Latvian language understanding. In: Human Language Technologies–the Baltic Perspective: Proceedings of the Ninth International Conference Baltic HLT 2020, vol. 328, p. 111.