EMBEDDIA

Cross-Lingual Embeddings for Less-Represented Languages in European News Media

Research and Innovation Action
Call: H2020-ICT-2018-1
Call topic: ICT-29-2018 A multilingual Next generation Internet
Project start: 1 January 2019 Project duration: 36 months

D1.10: Final evaluation report on cross-lingual embedding technology (T1.5)

Executive summary

In this deliverable, we present the final evaluation of embeddings and cross-lingual embeddings developed in WP1 of the EMBEDDIA project. The report focuses on the most successful contextual text representations: ELMo and BERT models. We evaluate variants of these models on a large collection of challenging monolingual and cross-lingual benchmarks in several languages: named entity recognition, POS tagging, dependency parsing, CoSimLex datasets, analogies, and SuperGLUE benchmarks. Our results show that contextual models developed within the EMBEDDIA project are superior in many tasks and enable a successful cross-lingual transfer of prediction models to less-resourced languages.

Partner in charge: UL

Project co-funded by the European Commission within Horizon 2020 Dissemination Level		
PU	Public	PU
PP	Restricted to other programme participants (including the Commission Services)	–
RE	Restricted to a group specified by the Consortium (including the Commission Services)	–
CO	Confidential, only for members of the Consortium (including the Commission Services)	–

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825153

Deliverable Information

Document administrative information	
Project acronym:	**EMBEDDIA**
Project number:	**825153**
Deliverable number:	**D1.10**
Deliverable full title:	**Final evaluation report on cross-lingual embedding technology**
Deliverable short title:	**Final embeddings evaluation**
Document identifier:	**EMBEDDIA-D110-FinalEmbeddingsEvaluation-T15-submitted**
Lead partner short name:	**UL**
Report version:	**submitted**
Report submission date:	**30/06/2021**
Dissemination level:	**PU**
Nature:	**R = Report**
Lead author(s):	**Matej Ulčar (UL), Marko Robnik-Šikonja (UL)**
Co-author(s):	**Aleš Žagar(UL), Matic Kavaš (UL), Matthew Purver (QMUL), Carlos S. Armendariz (QMUL), Andraž Repar(JSI), Senja Pollak (JSI)**
Status:	**__ draft, __ final, _x_ submitted**

The EMBEDDIA Consortium partner responsible for this deliverable has addressed all comments received. Changes to this document are detailed in the change log table below.

Change log

Date	Version number	Author/Editor	Summary of changes made
12/05/2021	v1.0	Marko Robnik-Šikonja (UL)	First draft.
21/05/2021	v1.1	Matej Ulčar (UL)	Adding to the report.
22/05/2021	v1.2	Aleš Žagar (UL)	Adding to the report.
23/05/2021	v1.3	Carlos S. Armandariz (QMUL)	Adding to the report.
24/05/2021	v1.4	Senja Pollak (JSI)	Adding to the report.
31/05/2021	v1.5	Matej Ulčar (UL)	Polishing the report.
31/05/2021	v1.6	Marko Robnik-Šikonja (UL)	Polishing the report.
05/06/2021	v1.7	Antoine Doucet (ULR)	Internal review.
07/06/2021	v1.8	Matej Martinc (JSI)	Internal review.
14/06/2021	v1.9	Marko Robnik-Šikonja (UL), Matej Ulčar (UL)	Revision based on the internal reviews.
15/06/2021	v1.10	Nada Lavrač (JSI)	Report quality checked and finalised.
20/06/2021	final	Marko Robnik-Šikonja (UL)	Final corrections.
30/06/2021	submitted	Tina Anžič (JSI)	Report submitted.

Table of Contents

List of abbreviations

BERT	Bidirectional Encoder Representations from Transformers
CBOW	Continuous Bag-of-Words
CNN	Convolutional Neural Network
DP	dependency parsing
ELMo	Embeddings from Language Models
GAN	Generative Adversarial Networks
GLUE	General Language Understanding Evaluation
HT	Human Translation
LAS	Labelled Attachment Score
LM	Language Model
LSTM	Long Short-Term Memory
MLM	Masked Language Model
MT	Machine Translation
NLP	Natural Language Processing
NLU	Natural Language Understanding
NLI	Natural Language Inference
NE	Named Entity
NEL	Named Entity Linking
NER	Named Entity Recognition
POS	Part of speech
QA	Question Answering
RoBERTa	Robust Bidirection Encoder Representations from Transformers
SuperGLUE	Super General Language Understanding Evaluation
UAS	Unlabelled Attachment Score

1 Introduction

The EMBEDDIA project aims to improve the cross-lingual transfer of language resources and trained models using word embeddings and cross-lingual word embeddings. The objectives of work package WP1 of the EMBEDDIA project are to advance cross-lingual and context-dependent word embeddings and test them with deep neural networks. This WP forms a technological basis for other WPs in the project, in particular WP3, WP4, and WP5 that work on concrete news media problems. To demonstrate advancements, EMBEDDIA covers English and eight less-resourced languages: Croatian, Estonian, Finnish, Latvian, Lithuanian, Russian, Slovene, and Swedish. The specific objectives of WP1 are as follows:

1. advance cross-lingual and multilingual word embeddings technology in T1.1,
2. advance context-dependent and dynamic embeddings technology in T1.2,
3. advance deep learning technology for morphologically rich, less-resourced languages in T1.3,
4. improve the interpretability of models and visualisation of results in T1.4,
5. collect and prepare datasets and benchmarks required to evaluate the developed technologies in T1.5.

This report titled *Final evaluation report on cross-lingual embedding technology* describes the results of the work performed in T1.5 from M10 to M30. It is the final deliverable of this task and of the whole WP1. The initial work within T1.1 from M1 to M9 was reported and accepted as Deliverable D1.1 in M9. That work covered the initial collection of datasets, benchmarks and evaluation metrics for word embeddings and cross-lingual embeddings, while in this report, we
i) describe further datasets and benchmarks not covered in the initial report, and
ii) report on the evaluation of the most successful embedding approaches developed within WP1 using collected datasets and benchmarks.

The main contributions presented in this report (in the order of appearance) are as follows.

1. Translation and adaptation of the SuperGLUE (Super General Language Understanding Evaluation) benchmark to Slovene (less-resourced language), presented in Section 3.2.5. This achievement allows cross-lingual evaluation of models on several important natural language understanding (NLU) tasks: question answering (QA), coreference resolution and natural language inference (NLI).
2. Evaluation of monolingual embedding approaches produced in WP1, presented in Section 4.1.
3. Evaluation of cross-lingual embedding approaches produced in WP1, presented in Section 4.2.

Besides these contributions, the work in T1.5 has contributed to the achievements reported in all other tasks of WP1. The presented evaluations will guide further research in workpackages WP3, WP4, and WP5. Furthermore, the produced resources are integrated into EMBEDDIA Media Assistant and ClowdFlows platform as contributions to WP6.

The structure of the report is as follows. In Section 2, we describe the used monolingual and cross-lingual approaches. We split them into four parts: baseline non-contextual fastText embeddings, contextual ELMo embeddings, cross-lingual maps for these two, and BERT-based monolingual and cross-lingual models. In Section 3, we present evaluation scenarios, divided into settings and benchmarks. Section 4 contains the results of the evaluations. We first cover the monolingual approaches, followed by the cross-lingual ones. The conclusions are presented in Section 5, and the outputs associated with this report are collected in Section 6.

2 Cross-lingual and contextual embedding

In this section, we shortly describe the used monolingual and cross-lingual approaches. Detailed descriptions of various methods are contained in previous Deliverables D1.2, D1.3, D1.6, and D1.7. In Section 2.1, we first present the non-contextual fastText baseline, and in Section 2.2, the contextual ELMo embeddings. Mapping methods for the embedding spaces produced by these two types of approaches are discussed in Section 2.3. We describe large pretrained language models based on the transformer neural network architecture in Section 2.4.

2.1 Baseline fastText embeddings

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

Mikolov et al. (2013) introduced the word2vec method and trained it on a huge Google News data set (about 100 billion words). The pretrained 300-dimensional vectors for 3 million English words and phrases are publicly available[1]. 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., ± 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 EMBEDDIA. 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:

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

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

2.2 ELMo embeddings

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

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

on the given following words. The second LSTM layer is equivalent to the first LSTM, just reading the text in reverse.

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

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

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

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

At the time of its introduction, ELMo has been shown to outperform previous pretrained word embeddings like word2vec and GloVe on many NLP tasks, e.g., question answering, named entity extraction, sentiment analysis, textual entailment, semantic role labelling, and coreference resolution (Peters et al., 2018). Later, BERT models turned out to be even more successful on these tasks. However, as our work shows, concerning the quality of extracted vectors, ELMo is often advantageous (Škvorc et al., 2020). The information it contains is condensed into only three layers, while multilingual BERT uses 14 layers. We reported on our novel cross-lingual mappings suitable for ELMo precomputed contextual models (Ulčar & Robnik-Šikonja, 2020) in Deliverable D1.6. The actual implementations of ELMo models result from our work in T1.2 (Ulčar & Robnik-Šikonja, 2020a), where we developed several ELMo contextual embeddings approaches for the languages covered in the EMBEDDIA project.

We compare EMBEDDIA ELMo models with models produced in the ELMoForManyLangs (Che et al., 2018) project. These models were trained on significantly smaller 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).

2.3 Cross-lingual maps for fastText and ELMo

Cross-lingual alignment methods take precomputed word embeddings for each language and align them with the optional use of bilingual dictionaries. The goal of alignments is that the embeddings for words with the same meaning shall be as close as possible in the final vector space. A comprehensive summary of existing approaches can be found in (Artetxe et al., 2018). In addition, we reported on the existing and newly proposed mapping techniques in Deliverable D1.6.

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

To align contextual ELMo embeddings, we developed two methods that take different contexts and word meanings into account (reported in Deliverable D1.6). Both methods require a different type of contextual mapping datasets, described in Section 2.3.1. The datasets map not only words but also their contexts. The first contextual mapping approach requires these datasets but uses the same isomorphic embedding mapping methods as commonly used for fastText embeddings (described in Section 2.3.2) for alignment of contextual embeddings. The second cross-lingual contextual mapping approach, described in Section 2.3.3, uses the same contextual datasets but drops the assumption that the aligned spaces are isomorphic.

2.3.1 Cross-lingual contextual dataset

The main obstacle to form a cross-lingual mapping between contextual embeddings is that a word in one language is represented with several different words (one for each meaning) in another language. We proposed two novel methods for the alignment of contextual embeddings based on the idea of matching contexts in different languages (see Section 2.3.2 and Section 2.3.3). For that, we require two resources: a sentence aligned parallel corpus of the two covered languages and their bilingual dictionary. The dictionary alone is not sufficient, as the words are not given in the context; therefore, it cannot help for 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).

We used the OpenSubtitles parallel corpus[2] (Lison & Tiedemann, 2016) from the Opus web page[3] for each pair of languages that we evaluated. The dictionaries we used are bilingual dictionaries extracted from wiktionary, using wikt2dict[4] tool (Acs, 2014). We extracted dictionaries for each EMBEDDIA language paired with English and the following language pairs of similar languages: Croatian-Slovenian, Estonian-Finnish, and Latvian-Lithuanian. For the language pairs not involving English, we created two different dictionaries, a direct bilingual dictionary and a dictionary created with triangulation via English. Dictionaries created with triangulation have more entries but are of worse quality than direct dictionaries. After the extraction, we manually cleaned the dictionaries using filters, such as removing accent marks on vowels from languages that do not use them (e.g., Slovenian) and removing extra non-alphabetical characters, like brackets, colon, and hash. We limited the dictionaries to entries with single-word terms in both languages.

2.3.2 Isomorphic maps

The first method we developed for the computation of cross-lingual mappings between contextual embeddings is based on the assumption that the aligned spaces are isomorphic. With a large enough collection of words in matching contexts, we compute their contextual embedding vectors and align them with any non-contextual mapping method. We use either the vecmap library (Artetxe et al., 2018), which showed the best performance in our experiments, reported in Deliverable D1.2, or the MUSE library (Conneau et al., 2018), which only aligns target vectors and is therefore computationally more efficient. To test this approach, we work with ELMo contextual embeddings due to their advantage over BERT concerning extracted vectors.

[2] https://www.opensubtitles.org/.

[3] http://opus.nlpl.eu

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

In our isomorphic method for alignment of ELMo contextual embeddings, we approached the creation of the contextual mapping dataset in two ways, one for contextual ELMo layers and the second for the non-contextual ELMo layer. The first of the three ELMo layers is non-contextual. We calculated the non-contextual part (i.e. the first layer) of ELMo embeddings for each pair of words in the bilingual dictionary. We used that as our list of non-contextual anchors.

For contextual ELMo layers, we lemmatised the parallel corpora using the Stanza tool (Qi et al., 2020). We then processed each corpus context by context. For each context, we calculated the embeddings of the non-lemmatised corpus. We then checked for each word of the lemmatised context if its pair from the bilingual dictionary appears in the same lemmatised context of the other language. When such a match was found, the two words' IDs and their contextual part of ELMo embeddings (i.e. the second and third layer) were added to the list of contextual anchors. The reason for the lemmatisation is that the bilingual dictionaries predominantly contain lemmas of the words. Note that we still use the non-lemmatised corpus in the computation of embeddings to get the correct contexts. In creating the contextual mapping dataset, we considered at most 20 different contexts of each lemma, not to overwhelm the dataset with frequent words (such as stop words).

We split the created datasets of anchor lists into the training and testing part. The training part takes 98.5% of the whole dataset for each language pair, and the testing part takes 1.5%. These datasets (one for each layer) were used to map one vector space to another, allowing us to map one word with multiple meanings in one language to multiple words in another language.

We used the computed bilingually aligned contextual embedding pairs as an input to methods that align two monolingual embeddings. To get the cross-lingual alignment, we used the vecmap supervised method (Artetxe et al., 2018) or the MUSE supervised method (Conneau et al., 2018).

2.3.3 Non-isomorphic maps

As several researchers have observed, the monolingual embedding spaces of two different languages are not completely isomorphic, which is especially true for distant languages (Ormazabal et al., 2019). This causes error in methods that assume isomorphism of embedding spaces, including the commonly used vecmap and MUSE methods.

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) are a type of neural networks consisting of two connected neural models, a generator and a discriminator. The two models are trained simultaneously via an adversarial process. The discriminator attempts to discern whether the data passed to its input is real or fake (i.e. artificially generated). At the same time, the generator attempts to generate artificial data, which can fool the discriminator. GANs play a zero-sum game, where the success of the discriminator means a failure of the generator and vice versa. By simultaneously training both networks, they both improve. GANs are primarily used on images, where the described process can lead to compelling new generated images.

Following the success of GANs in neural machine translation (Yang et al., 2018) and unsupervised cross-lingual alignment (Conneau et al., 2018; Fu et al., 2020), we proposed a novel supervised non-linear mapping method using bidirectional GANs. We based our contextual alignment method, called ELMo-GAN, on the model of Fu et al. (2020). Contrary to Fu et al. (2020), who only used their method with non-contextual fastText embeddings (Bojanowski et al., 2017) to align sentences, we align contextual ELMo embeddings (Peters et al., 2018), which is only possible by constructing special contextual mapping datasets, described in Deliverable D1.6.

The GAN mapping comprises the generator module and the discriminator module. The generator module contains two generators that map vectors from one vector space to the other. Specifically, for a pair of languages L_1 and L_2, one generator will map from L_1 to L_2, and the second will map from L_2 to L_1. The discriminator module contains two discriminators. The first discriminator tries to predict whether a given pair of vectors represent the same token, i.e. if the first vector represents the word x in L_1 and the second vector represents the translation of the word x in L_2. The second discriminator attempts to learn the difference between the direction of mapping. For a given pair of vectors, it predicts whether they are a vector from L_1 and its mapping to L_2 or a vector from L_2 and its mapping to L_1.

We produced two different versions of the ELMoGAN based on the number of iterations in the model's training. The first version (ELMoGAN-10k) was trained for a fixed number of $10\,000$ iterations for each layer of each language pair. The second version (ELMoGAN-O) was trained for the number of iterations that gave the best result in the dictionary induction task, using the evaluation dictionary. This choice was determined in our preliminary tests on unrelated NER tasks and might not be optimal for other tasks.

2.4 BERT embeddings

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

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

BERT has shown excellent performance on 11 NLP tasks: 8 from GLUE language understanding benchmark (Wang et al., 2018), question answering, named entity recognition, and common-sense inference (Devlin et al., 2019). The performance on monolingual tasks has often improved upon ELMo. However, while multilingual BERT covers 104 languages, its subword dictionary comprises tokens from all covered languages, which might not be optimal for a particular language. Further, similarly to ELMo, its training and tuning are computationally highly demanding tasks out of reach for most researchers.

2.4.1 Monolingual and massively multilingual BERT and RoBERTa models

The original BERT project offers pretrained English, Chinese, Spanish, and multilingual models. The multilingual BERT model (mBERT) is trained simultaneously on 104 languages, including all EMBEDDIA languages, using vast amounts of data. The mBERT model provides a representation in which the languages are embedded in the same space without requiring further explicit cross-lingual mapping. This massively multilingual representation might be sub-optimal for any specific language or subset of languages.

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

An open source DeepPavlov library[5] offers a specific Russian BERT model. Recently, monolingual Finnish (FinBERT) (Virtanen et al., 2019), Estonian (EstBERT) (Tanvir et al., 2020), Latvian (LVBERT)

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

(Znotiņš & Barzdiņš, 2020), and Swedish (KB-BERT) (Malmsten et al., 2020) BERT models were released.

The quality of these monolingual BERT models vary. For some languages, EMBEDDIA internal datasets are large and of high-quality. For these languages, we expected to train a better monolingual model (Estonian) or train a non-existent high-quality monolingual model (Slovene). We describe these models in the next paragraph. For pairs of similar languages, we trained trilingual BERT models in combination with English (see Section 2.4.2). We did not train models for Swedish or Russian, as the training procedure is computationally intensive, and we do not have enough high-quality data to improve over already existing monolingual models in these two languages.

We have trained two monolingual RoBERTa-based models, one on Slovene (SloBERTa) and one on Estonian (Est-RoBERTa). 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 our 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 is composed of general language corpus, web-crawled texts, academic writings (BSc/BA, MSc/MA and PhD theses) and texts from Slovenian parliament. Est-RoBERTa was trained for about 40 epochs on Estonian corpora, containing mostly news articles from Express Meedia. The corpora has 2.51 billion tokens in total. We used the sentencepiece algorithm[6] to produce subword byte-pair-encodings (BPE) from a given training dataset. The created subword vocabularies contain 32,000 tokens for SloBERTa model and 40,000 tokens for Est-RoBERTa model. Both models are publicly available via the popular Hugging Face library and for individual download from CLARIN (see Section 6).

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

2.4.2 Trilingual EMBEDDIA BERT models

At the start of this task, there were no language-specific BERT models for EMBEDDIA languages other than English (later, the Russian version mentioned above appeared). Therefore, we trained new BERT models for EMBEDDIA languages, as presented in Deliverable D1.7. We decided to build trilingual models featuring two similar languages and one highly resourced language (English). Because these models are trained on a small number of languages, they better capture each of them and offer better monolingual performance. At the same time, they can be used in a cross-lingual manner for knowledge transfer from a high-resource language to a low-resource language or between similar languages.

We have trained three trilingual models, one on Slovene, Croatian and English data (CroSloEngual BERT), one on Estonian, Finnish and English (FinEst BERT), and one on Latvian, Lithuanian and English (LitLat BERT). The models are now publicly available via the popular Huggingface library and for individual download from CLARIN (see Section 6). For each model, we combined deduplicated corpora from all three languages. The corpora used to train our BERT models are described in Deliverable D1.7.

FinEst BERT and CroSloEngual BERT were trained on BERT-base architecture (Ulčar & Robnik-Šikonja,

[6] https://github.com/google/sentencepiece

2020). We used bert-vocab-builder[7] to produce wordpiece vocabularies (composed of subword tokens) from the given corpora. The created wordpiece vocabularies contain 74,986 tokens for FinEst and 49,601 tokens for the CroSloEngual model. The training dataset is a masked corpus. We randomly masked 15% of the tokens in the corpus and repeated the process five times, each time with different 15% of the tokens being masked. The dataset is thus five times larger than the original corpora. On this data, we trained our BERT models for about 40 epochs, which is approximately the same as multilingual BERT.

Later LitLat BERT is based on the RoBERTa architecture. We opted for the RoBERTa approach because it has since proven more robust and better performing than BERT. It also offered two practical benefits over the original BERT approach. By dropping the next-sentence prediction training task, corpora shuffled on the sentence level can be used in training at the expense of more limited context (compared to the original 512 tokens used in BERT). The second benefit is that it allows for training on multiple GPUs out of the box, while BERT can only be trained on a single GPU unless complex workarounds are implemented. We split the Lithuanian, Latvian and English corpora into three sets, train, eval and test. Train dataset contains 99% of all the corpora; the other two sets contain 0.5% each. We used the sentencepiece algorithm[8] to produce subword byte-pair-encodings (BPE) from a given train dataset. The created subword vocabulary contains 84,200 tokens. We have trained the model for 40 epochs, with a maximum sequence length of 512 tokens. Like with FinEst BERT and CroSloEngual BERT, we randomly masked 15% of the tokens during the training.

[7] https://github.com/kwonmha/bert-vocab-builder

[8] https://github.com/google/sentencepiece

3 Evaluation scenarios

In this section, we describe the evaluation scenarios. First, in Section 3.1, we describe the settings of monolingual and cross-lingual evaluation experiments. In Section 3.2, we describe the datasets.

3.1 Evaluation settings

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

3.1.1 Named entity recognition

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

For BERT models, we fine-tuned each model on the NER dataset for 3 epochs. We used the code by HuggingFace[9] for NER classification.

3.1.2 POS-tagging

For training POS-tagging classifiers with fastText or ELMo embeddings, we used the same approach and hyper-parameters as described above for NER, but a different neural network architecture. We trained models with four hidden layers, three bidirectional LSTMs and one fully connected feed-forward layer. The three LSTM layers have 512, 512, and 256 units, respectively. The fully connected layer has 64 neurons.

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

3.1.3 Dependency parsing

To train dependency parsers using ELMo embeddings, we used SuPar tool by Yu Zhang.[10] SuPar is based on the deep biaffine attention (Dozat & Manning, 2017). We modified the SuPar tool to accept ELMo embeddings on the input; specifically, we used the concatenation of the three ELMo vectors. The modified code has been made publicly available (see Section 6). We trained the parser for 10 epochs for each language, using separately EMBEDDIA ELMo embeddings and ELMoForManyLangs embeddings.

For training BERT models, we modified the dep2label-bert tool (Strzyz et al., 2019; Gómez-Rodríguez et al., 2020) to work with newer versions of HuggingFace's transformers library and to support both

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

[10] https://github.com/yzhangcs/parser

RoBERTa and BERT-based models. We used the modified tool to fine-tune all the BERT/RoBERTa models on the dependency parsing task for 10 epochs. We used arc-Standard algorithm in transition-based sequence labelling encoding. The modified tool is publicly available (see Section 6).

3.1.4 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. We used a boilerplate sentence "If the term [w1] corresponds to the term [w2], then the term [w3] corresponds to the term [w4]." Here, [w1] through [w4] represent the four words from an analogy entry. We translated the boilerplate sentence to every other EMBEDDIA language for evaluation in those languages.

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

BERT models are masked language models, so we tried to exploit that in this task. We masked the word "w2" and tried to predict it, given every other word. In cross-lingual setting the sentence after the comma and the words "w3" and "w4" were therefore given in the source/training language, while the sentence before the comma and word "w1" were given in target/evaluation language. The prediction for masked word "w2" was expected in the target/testing language, as well.

3.1.5 SuperGLUE

We fine-tuned BERT models on SuperGLUE tasks using Jiant tool (Phang et al., 2020). We used a single-task learning setting for each task and fine-tuned for 100 epochs, 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 evaluation of SuperGLUE tasks and the tasks themselves are described in more detail in Section 3.2.5 and Section 4.1.6.

3.2 Datasets

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

3.2.1 Named entity recognition

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

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

Language	Dataset	Sentences	Tags
Croatian	hr500k	24794	28902
English	CoNLL-2003 NER	20744	43979
Estonian	Estonian NER corpus	14287	20965
Finnish	FiNER data	14484	16833
Latvian	LV Tagger train data	9903	11599
Lithuanian	TildeNER	5500	7000
Russian	factRuEval-2016	4907	9666
Slovene[11]	ssj500k	9489	9440
Swedish	Swedish NER	9369	7292

3.2.2 POS-tagging and Dependency parsing

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

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

Language	Treebank	Tokens	Sentences	Train	Validation	Test
Croatian	SET	197044	8889	6983	849	1057
English	EWT	254854	16622	12543	2002	2077
Estonian	EDT	434245	30723	24384	3125	3214
Finnish	TDT	202208	15136	12217	1364	1555
Latvian	LVTB	152706	9920	7163	1304	1453
Lithuanian	HSE	5356	263	153	55	55
Russian	GSD	99389	5030	3850	579	601
Slovene	SSJ	140670	8000	6478	734	788
Swedish	Talbanken	96858	6026	4303	504	1219

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

3.2.3 CoSimLex

In contrast to other datasets which are used to evaluate the performance of embeddings on specific tasks, the CoSimLex task (Armendariz et al., 2020), described in Deliverable D1.3, allows direct investigation of embeddings' properties. CoSimLex contains pairs of words and their similarity ratings assigned by human annotators. The crucial difference to previous such datasets is that the words appeared within a short text (context) when presented to the human annotators. Therefore, the word

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

similarity ratings take the context into account, making the dataset suitable to evaluate the contextualised embeddings. Furthermore, 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. For Croatian and Finnish, we used the existing translations of SimLex-999 (Mrkšić et al., 2017; Venekoski & Vankka, 2017; Kittask, 2019). For Slovene, T1.2 produced a new translation, following Mrkšić et al. (2017)'s methodology for Croatian; this has now been made publicly available[12].

For each pair of words, two different contexts are extracted from Wikipedia 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 annotator ratings for that context. This is accompanied by two standard deviation scores, the p-value computed from the Mann-Whitney U test on the two score distributions, and the four inflected forms of the words exactly as they appear in the contexts. Note that in the morphologically rich languages (such as Slovene, Croatian, and Finnish), many inflections are possible. CoSimLex is the only reasonably sized dataset in which differences in the contextual similarity between two words are supported with a test of statistical significance. Figure 1 shows an example from the English dataset.

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

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

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

1 - 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. We use the uncentered Pearson correlation. This gives a measure of the accuracy of predicting the relative magnitude of changes and allows for differences in scaling while maintaining the effect of the direction of change. The standard centered correlation normalises on the mean, so it could give high values even when a system predicts changes in the wrong direction, but with a similar distribution over examples.

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

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

[12] http://hdl.handle.net/11356/1309

3.2.4 Monolingual and cross-lingual analogies

The word analogy task was popularised by Mikolov et al. (2013). The goal is to find a term y for a given term x so that the relationship between x and y best resembles the given relationship $a : b$. 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, for example, adjective degrees of comparison. The two words in any given pair then have a common stem (or even the same lemma); for example, given the word pair "long : longer", we see that we have an adjective in its base form and the same adjective in a comparative form. The task is then to find the term y corresponding to the relationship "dark : y", with the expected answer being $y =$ darker, i.e. a comparative form of the adjective dark.

In the vector space, the analogy task is transformed into vector arithmetic. We search for nearest neighbours, i.e. we compute the distance between vectors: d(vec(Finland), vec(Helsinki)) and search for word y which would give the closest result in distance d(vec(Sweden), vec(y)). In our datasets, created in T1.1 (Ulčar et al., 2020), 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 statistics called accuracy@5, which we report as a result.

In the cross-lingual setting between two languages L_1 and L_2, the word analogy task (x is to y as a is to b) using a cross-lingual dataset is composed by matching each relation in one language with each relation from the same category in the other language. Unfortunately, for cross-lingual contextual mappings, the word analogy task is not adequate as it only contains words without their context. We described our approach for applying this task in Section 3.1.4.

3.2.5 SuperGLUE tasks

SuperGLUE (Super General Language Understanding Evaluation) (Wang et al., 2019) is a benchmark for testing natural language understanding (NLU) of models. It is styled after the GLUE benchmark (Wang et al., 2018), but much more challenging. It provides a single-number metric for each of its tasks that enables the comparison and progress of NLP models. The tasks are diverse and comprised of question answering (BoolQ, COPA, MultiRC, and ReCoRD tasks), natural language inference (CB and RTE tasks), coreference resolution (WSC), and word sense disambiguation (WiC). Non-expert humans evaluated all the tasks to give a human baseline to machine systems. We provide an example of each task in Table 3. Please refer to the original paper for an extensive description of each task.

To evaluate cross-lingual transfer and test specifics of morphologically rich languages, we translated the SuperGLUE datasets to Slovene. We partially used human translation (HT) and partially machine translation (MT). The details are presented in Table 4. Some datasets are too large (BoolQ, MultiRC, ReCoRD, RTE) to be fully human translated with our budget. We thus provide ratios between the human translated and the original English sizes. For MT from English to Slovene, we used the GoogleMT Cloud service. In our evaluation, we use six of the original eight tasks.

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

We did not include ReCoRD in the Slovene benchmark due to the low quality of our test set, consisting of confusing and ambiguous examples. Further, there are differences between English and Slovene ReCoRD tasks due to the morphological richness of Slovene. Namely, in Slovene, the correct declen-

Table 3: Examples from the development set of SuperGLUE tasks. **Bold** texts represent parts of examples' format. Texts in *italics* are part of models' input. Underlined texts are specially marked in inputs. Texts in the `monospaced font` represent the expected models' outputs.

BoolQ	**Passage**:*Barq's – Barq's is an American soft drink. Its brand of root beer is notable for having caffeine. Barq's, created by Edward Barq and bottled since the turn of the 20th century, is owned by the Barq family but bottled by the Coca-Cola Company. It was known as Barq's Famous Olde Tyme Root Beer until 2012.* **Question**:is barq's root beer a pepsi product **Answer**:`No`
CB	**Text**:*B: And yet, uh, I we-, I hope to see employer based, you know, helping out. You know, child, uh, care centers at the place of employment and things like that, that will help out. A: Uh-huh. B: What do you think, do you think we are, setting a trend?* **Hypothesis**:they are setting a trend **Entailment**:`Unknown`
COPA	**Premise**:*My body cast a shadow over the grass.* **Question**:What's the CAUSE for this **Alternative 1**:The sun was rising. **Alternative 2**:The grass was cut. **Correct Alternative**:`1`
MultiRC	**Paragraph**:*Susan wanted to have a birthday party. She called all of her friends. She has five friends. Her mom said that Susan can invite them all to the party. Her first friend could not go to the party because she was sick. Her second friend was going out of town. Her third friend was not so sure if her parents would let her. The fourth friend said maybe. The fifth friend could go to the party for sure. Susan was a little sad. On the day of the party, all five friends showed up. Each friend had a present for Susan. Susan was happy and sent each friend a thank you card the next week.* **Question**:Did Susan's sick friend recover? **Candidate answers**: Yes, she recoverd, No (`F`), Yes (`T`), No, she didn't recover (`F`), Yes, she was at Susan's party (`T`)
ReCoRD	**Paragraph**:*(CNN) Puerto Rico on Sunday overwhelmingly voted for statehood. But Congress, the only body that can approve new states, will ultimately decide whether the status of the US commonwealth changes. Ninety-seven percent of the votes in the nonbinding referendum favored statehood, an increase over the results of a 2012 referendum, official results from the State Electoral Commission show. It was the fifth such vote on statehood. Today, we the people of Puerto Rico are sending a strong and clear message to the US Congress ... and to the world ... claiming our equal rights as American* citizens, *Puerto Rico* Gov. *Ricardo Rossello said in a news release.@highlight Puerto Rico voted Sunday in favor of US statehood* **Query**:For one, they can truthfully say, "Don't blame me, I didn't vote for them," when discussing the ⟨placeholder⟩ presidency. **Correct Entities**: `US`
RTE	**Text**:*Dana Reeve, the widow of the actor Christopher Reeve, has died of lung cancer at age 44, according to the Christopher Reeve Foundation.* **Hypothesis**:Christopher Reeve had an accident. **Entailment**: `False`
WiC	**Context 1**:*Room and board.* **Context 2**:He nailed boards across the windows. **Sense match**: `False`
WSC	**Text**:*Mark told Pete many lies about himself, which Pete included in his book. He should have been more truthful.* **Coreference**: `False`

sion of a query is 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 (the example in Table 3 demonstrates that, i.e. the word *board* in two different contexts translates to two completely different words in Slovene).

Table 4: 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	110	0.4400	yes
	val	57	22	0.3860	yes
	test	250	110	0.4400	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	25	0.0139	yes
ReCoRD	train	101000	60	0.0006	/
	val	10000	6	0.0006	/
	test	10000	30	0.0030	/
RTE	train	2500	232	0.0928	yes
	val	278	29	0.1043	yes
	test	300	29	0.0967	yes
WiC	train	6000	/	/	/
	val	638	/	/	/
	test	1400	/	/	/
WSC	train	554	554	1.0000	/
	val	104	104	1.0000	/
	test	146	146	1.0000	/

4 Results

We present two sets of results. First, in Section 4.1 we evaluate monolingual models, followed by evaluation of cross-lingual transfer in Section 4.2.

4.1 Monolingual evaluations

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

4.1.1 NER

In Table 5, we present the results of fastText non-contextual baseline, compared with two types of contextual ELMo embeddings, ELMoForManyLanguages and EMBEDDIA ELMo. EMBEDDIA ELMo, trained on much larger datasets, is the best on every language except Latvian. The fastText baseline lags behind both ELMo embeddings. For English, we use the original ELMo model, as EMBEDDIA ELMo does not exist for English. This model is also better than the ELMoForManyLanguages.

Table 5: The comparison of fastText non-contextual baseline with two types of ELMo embeddings (ELMoForManyLanguages - EFML and EMBEDDIA ELMo) on the NER task. The results are given as macro F_1 scores. The best model for each language is in **bold**. For English (marked with *), we show the original ELMo model.

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

The results of BERT models are presented in Table 6. Each of the listed BERT models was fine-tuned on NER datasets in languages where that makes sense: monolingual and trilingual BERT models were used in languages used in their pretraining, and massively multilingual models (mBERT and XLM-R) were fine-tuned for all used languages.

The results show that EMBEDDIA BERT models are very successful, dominating in all languages where they exist. The two EMBEDDIA monolingual models (Slovene and Estonian) have a slight edge over their trilingual counterparts. Comparing BERT results in Table 6 with ELMo results in Table 5, we can observe clear dominance of BERT models. The extracted ELMo embedding vectors are clearly not competitive to the entire pretrained BERT models.

4.1.2 POS-tagging

In Table 7, we present the results of fastText non-contextual baseline, compared with two types of contextual ELMo embeddings, ELMoForManyLanguages and EMBEDDIA ELMO. Again, EMBEDDIA ELMo models and DeepPavlov ELMo for Russian (trained on much larger datasets) are the best on all languages. Some results are surprisingly low but this is the effect of low quality of the datasets.

Table 6: The results of NER evaluation task for various BERT models. The scores are macro average F_1 scores of the three NE classes. We compare BERT models produced outside and inside the EMBEDDIA project. The non-EMBEDDIA models are massively multilingual models mBERT and XLM-R, as well as monolingual (MONO) models (FinBERT for Finnish, EstBERT for Estonian, LVBERT for Latvian, BERTić for Croatian, bert-base-cased for English, KB-BERT for Swedish). EMBEDDIA trilingual BERT models are CroSloEngual BERT (CSE), FinEst BERT, and LitLat BERT. EMBEDDIA monolingual BERT models (E-MONO) are SloBERTa for Slovene and Est-RoBERTa for Estonian.

	Non-EMBEDDIA			EMBEDDIA			
Language	mBERT	XLM-R	MONO	CSE	FinEst	LitLat	E-MONO
Croatian	0.801	0.833	0.881	**0.886**	-	-	-
English	0.938	0.941	0.943	**0.944**	0.937	0.939	-
Estonian	0.900	0.913	0.870	-	0.930	-	**0.936**
Finnish	0.934	0.932	0.952	-	**0.957**	-	-
Latvian	0.847	0.859	0.145	-	-	**0.863**	-
Lithuanian	0.833	0.802	-	-	-	**0.863**	-
Slovenian	0.885	0.912	-	0.928	-	-	**0.933**
Swedish	0.844	0.875	**0.887**	-	-	-	-

Table 7: The comparison of fastText non-contextual baseline with two types of ELMo embeddings (ELMoForManyLanguages - EFML and EMBEDDIA ELMo) on the POS-tagging task. The results are given as micro F_1 scores. The best results for each language are in **bold**. For English, the result of the original ELMo model is shown, and for Russian the results of DeepPavlov ELMo model (both marked with *).

Language	fastText	EFML	EMBEDDIA ELMo
Croatian	0.512	0.573	**0.963**
English	0.769	0.603	**0.952***
Estonian	0.640	0.508	**0.969**
Finnish	0.506	0.389	**0.966**
Latvian	0.462	0.489	**0.940**
Lithuanian	0.209	N/A	**0.233**
Russian	0.518	0.349	**0.929***
Slovenian	0.527	0.541	**0.966**
Swedish	0.275	0.313	**0.933**

Table 8: The results of POS-tagging evaluation task for various BERT models expressed with F_1 scores. We compare BERT models produced outside and inside the EMBEDDIA project. The non-EMBEDDIA models are massively multilingual models mBERT and XLM-R, as well as monolingual (MONO) models (FinBERT for Finnish, EstBERT for Estonian, LVBERT for Latvian, BERTić for Croatian, RuBERT for Russian, bert-base-cased for English, KB-BERT for Swedish). EMBEDDIA trilingual BERT models are CroSloEngual BERT (CSE), FinEst BERT, and LitLat BERT. EMBEDDIA monolingual BERT models (E-MONO) are SloBERTa for Slovene and Est-RoBERTa for Estonian. The best results for each language are in **bold**.

	Non-EMBEDDIA			EMBEDDIA			
Language	mBERT	XLM-R	MONO	CSE	FinEst	LitLat	E-MONO
Croatian	0.978	0.981	0.981	**0.982**	-	-	-
English	0.964	**0.972**	0.967	0.968	0.967	0.968	-
Estonian	0.966	0.970	0.961	-	0.973	-	**0.977**
Finnish	0.961	0.977	**0.980**	-	0.976	-	-
Latvian	0.946	0.960	0.048	-	-	**0.966**	-
Lithuanian	**0.855**	0.842	-	-	-	0.790	-
Russian	0.974	**0.976**	0.975	-	-	-	-
Slovenian	0.984	0.988	-	0.990	-	-	**0.991**
Swedish	0.979	0.981	**0.988**	-	-	-	-

The results of BERT models are presented in Table 8. Each of the listed BERT models was fine-tuned on POS datasets in languages where that makes sense: monolingual and trilingual BERT models were used in languages used in their pretraining, and massively multilingual models (mBERT and XLM-R) were fine-tuned for all used languages.

The results show that EMBEDDIA BERT models and massively multilingual BERT models are very competitive in the POS-tagging task, differences being relatively small and language-dependent. Nevertheless, for some languages the same pattern appears as in NER: the two EMBEDDIA monolingual models (Slovenian and Estonian) are the best again, the trilingual LitLat BERT is better then the monolingual Latvian BERT model, which was pretrained on insufficient amounts of data. Comparing BERT and ELMo results in Tables 7 and Table 8, we again observe a clear dominance of BERT models.

4.1.3 Dependency parsing

In Table 9, we compare two types of contextual ELMo embeddings, ELMoForManyLanguages and EMBEDDIA ELMo, on the dependency parsing task. EMBEDDIA ELMo models are the best on all languages where they exist. For Russian and English, ELMo models do not exist and we tested the original English ELMo and Russian DeepPavlov ELMo (marked with *), which are both better than ELMoForManyLanguages.

Table 9: The comparison of two types of ELMo embeddings (ELMoForManyLanguages - EFML and EMBEDDIA ELMo) on the dependency parsing task. Results are given as UAS and LAS scores. The best results for each language are typeset in **bold**. For English, the result of the original ELMo model is shown, and for Russian the result of DeepPavlov ELMo (both marked with *). There is no Lithuanian ELMoForManyLangs model.

	ELMoForManyLangs		EMBEDDIA ELMo	
Language	UAS	LAS	UAS	LAS
Croatian	88.18	79.45	**91.74**	**85.84**
English	90.28	86.29	**90.53***	**87.16***
Estonian	81.19	72.50	**89.54**	**85.45**
Finnish	88.27	83.44	**90.83**	**86.86**
Latvian	87.17	80.76	**88.85**	**82.82**
Lithuanian	-	-	**55.05**	**24.39**
Russian	89.28	83.29	**89.33***	**83.54***
Slovenian	85.55	77.73	**93.70**	**91.39**
Swedish	88.03	83.09	**89.70**	**85.07**

The results of BERT models are presented in Table 10. Each of the listed BERT models was fine-tuned on POS datasets in languages where that makes sense: monolingual and trilingual BERT models were used in languages used in their pretraining, and massively multilingual models (mBERT and XLM-R) were fine-tuned for all used languages.

The results show that the differences between EMBEDDIA BERT models and massively multilingual BERT models are language-dependent. Surprisingly, comparing BERT and ELMo results in Tables 9 and Table 10, shows that EMBEDDIA ELMo models dominate in all languages. These results indicate that BERT models shall not always be the blind choice in text classification, as ELMo might still be competitive in some tasks.

4.1.4 CoSimLex

In Table 11, we present the comparison between different ELMo models (ELMoForManyLanguages and EMBEDIA ELMo) and different BERT models (massively multilingual mBERT and two EMBEDDIA

Table 10: The results of the evaluation in the dependency parsing task for various BERT models. The results are given as LAS scores. We compare BERT models produced outside and inside the EMBEDDIA project. The non-EMBEDDIA models are massively multilingual models mBERT and XLM-R, as well as monolingual (MONO) models (FinBERT for Finnish, EstBERT for Estonian, LVBERT for Latvian, BERTić for Croatian, RuBERT for Russian, bert-base-cased for English, KB-BERT for Swedish). EMBEDDIA trilingual BERT models are CroSloEngual BERT (CSE), FinEst BERT, and LitLat BERT. EMBEDDIA monolingual BERT models (E-MONO) are SloBERTa for Slovene and Est-RoBERTa for Estonian. The best results for each language are typeset in **bold**.

	Non-EMBEDDIA			EMBEDDIA			
Language	mBERT	XLM-R	MONO	CSE	FinEst	LitLat	E-MONO
Croatian	70.38	78.39	-	**82.36**	-	-	-
English	83.19	**84.94**	83.55	83.91	83.18	81.80	-
Estonian	56.27	68.91	77.44	-	75.67	-	**78.64**
Finnish	57.22	71.12	**83.64**	-	79.96	-	-
Latvian	54.61	69.26	56.87	-	-	**74.32**	-
Lithuanian	18.30	**24.34**	-	-	-	18.30	-
Russian	70.00	73.47	**80.90**	-	-	-	-
Slovenian	68.08	79.27	-	**85.38**	-	-	84.41
Swedish	74.04	80.93	**85.83**	-	-	-	-

trilingual models: CroSloEngual BERT and FinEst BERT. The performance is expressed with two metrics, uncentered Spearman correlation between the predicted and actual change of similarity scores (M1) and the harmonic mean of the Spearman and Pearson correlations between predicted and actual similarity scores (M2).

Table 11: The comparison of different ELMo (upper part) and BERT embeddings (lower part) on CoSimLex datasets. We compare the performance with the uncentered Spearman correlation between the predicted and true change of similarity scores (M1), and the harmonic mean of the Spearman and Pearson correlations between predicted and true similarity scores (M2). The best scores for each language and type of models are in **bold**.

	English		Croatian		Slovene		Finnish	
Model	M1	M2	M1	M2	M1	M2	M1	M2
ELMoForManyLangs	0.556	0.449	0.520	0.433	0.467	0.328	0.403	0.403
EMBEDDIA ELMo	**0.570**	**0.510**	**0.662**	**0.529**	**0.550**	**0.516**	**0.452**	**0.407**
mBERT	0.713	0.573	0.587	0.443	0.603	0.516	0.671	0.289
EMBEDDIA CroSloEng BERT	**0.719**	**0.601**	**0.715**	**0.642**	**0.673**	**0.589**	-	-
EMBEDDIA FinEst BERT	0.692	0.591	-	-	-	-	**0.672**	**0.533**

Among ELMo models, EMBEDDIA ELMo models dominate ELMoForManyLanguages, producing closer scores to humans in both metrics and all four languages. Among BERT models, both EMBEDDIA trilingual models dominate. Comparing ELMo and BERT models, BERT models are more successful and predict similarities closer to human assigned scores.

4.1.5 Analogies

Our previous work in Deliverable D1.3 has shown a clear advantage of BERT models over ELMo embeddings. For that reason, in this section, we only compare different BERT models. The results are presented in Table 12.

The results show that EMBEDDIA BERT models are strongly dominating in all languages where they exist. Furthermore, the two EMBEDDIA monolingual BERT models (Slovene and Estonian) have a considerable edge over their trilingual counterparts.

Table 12: The results of the word analogy task for various BERT models expressed as Accuracy@5. We compare BERT models produced outside and inside the EMBEDDIA project. The non-EMBEDDIA models are massively multilingual models mBERT and XLM-R, as well as monolingual (MONO) models (FinBERT for Finnish, EstBERT for Estonian, LVBERT for Latvian, BERTić for Croatian, RuBERT for Russian). EMBEDDIA trilingual BERT models are CroSloEngual BERT (CSE), FinEst BERT, and LitLat BERT. EMBEDDIA monolingual BERT models (E-MONO) are SloBERTa for Slovene and Est-RoBERTa for Estonian. The best results for each language are typeset in **bold**.

	Non-EMBEDDIA			EMBEDDIA			
Language	mBERT	XLM-R	MONO	CSE	FinEst	LitLat	E-MONO
Croatian	0.090	0.138	-	**0.278**	-	-	-
English	0.404	0.413	0.114	0.390	**0.439**	0.418	-
Estonian	0.093	0.251	0.165	-	0.224	-	**0.393**
Finnish	0.067	0.208	0.173	-	**0.285**	-	-
Latvian	0.026	0.118	0.118	-	-	**0.170**	-
Lithuanian	0.036	0.107	-	-	-	**0.214**	-
Russian	0.102	**0.189**	0.000	-	-	-	-
Slovenian	0.061	0.146	-	0.195	-	-	**0.409**
Swedish	0.052	0.097	**0.239**	-	-	-	-

4.1.6 SuperGLUE tasks

SuperGLUE benchmark is extensively used to compare large pretrained models in English[13]. In contrast to that, we concentrate on the Slovene translation of the SuperGLUE tasks, described in Section 3.2.5. Experiments in English have shown that ELMo embeddings are not competitive to pretrained transformer models like BERT in GLUE benchmarks (Wang et al., 2019). For this reason, we skip ELMo models and compare three BERT models in our experiments: monolingual Slovene SloBERTa, trilingual CroSloEngual BERT, and massively multilingual mBERT (bert-base-multilingual-cased[14]). Each model was fine-tuned using either MT or HT datasets of the same size. Only the translated content varies between both translation types; otherwise, they contain exactly the same examples. The splits of instances into train, validation and test sets is the same as in the English variant (but mostly considerably smaller, see Table 4).

In our analysis, we vary the sizes of datasets, translation types, and prediction models. Table 13 shows the results together with several baselines trained on the original English datasets. Some comparisons to English baselines are not fair 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 datasets, the only fair comparison is possible with the COPA and WSC. For the CB task, only half of the dataset is human translated (further human translation is planned).

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., 2019).

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 2.3 points. The only task where MT is better than HT is BoolQ using mBERT. The problem here might be the small size of the testing set (only 30 examples). We speculate that for challenging tasks such as the ones collected in the SuperGLUE benchmark, MT is not yet competitive to HT.

Considering the Avg scores in Table 13, CroSloEngual is the best performing model. However, we

[13] https://super.gluebenchmark.com/leaderboard
[14] https://huggingface.co/bert-base-multilingual-cased

Table 13: The SuperGLUE benchmarks in English (upper part) and Slovene (lower part). All English results are taken from (Wang et al., 2019). 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 Avg scores for each language are underlined.

Task	**Avg**	**BoolQ**	**CB**	**COPA**	**MultiRC**	**ReCoRD**	**RTE**	**WiC**	**WSC**
Models/Metrics		**Acc.**	**F1/Acc.**	**Acc.**	**$F1_a$/EM**	**F1/EM**	**Acc.**	**Acc.**	**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 (Slovene)	49.7	63.3	23.0/52.7	50.0	**77.3/0.3**	-	58.6	-	65.8
HT-mBERT	51.7	63.3	37.5/58.2	54.2	56.6/12.8	-	58.6	-	61.6
MT-mBERT	52.9	**70.0**	36.4/59.1	54.4	54.5/12.5	-	58.6	-	-
HT-CroSloEngual	56.9	63.3	46.6/67.3	58.2	52.0/8.7	-	**75.9**	-	56.2
MT-CroSloEngual	52.3	63.3	34.4/56.4	55.0	51.6/12.8	-	65.5	-	-
HT-SloBERTa	56.2	63.3	**53.3/69.1**	**61.8**	52.9/12.5	-	62.1	-	**73.3**
MT-SloBERTa	53.0	63.3	42.7/60.0	58.2	55.4/11.5	-	58.6	-	-
HT-Avg	55.0	63.3	**55.3**	**58.1**	32.6	-	**65.5**	-	63.7
MT-Avg	52.7	**65.5**	48.2	55.9	**33.1**	-	60.1	-	-

should be cautious with this conclusion. Recall that Avg scores do not include the WSC dataset, which was only human translated (MT is not possible for WSC). If we include the WSC dataset into the Avg calculation, the final Avg score is 56.8 for CroSloEngual and 59.1 for SloBERTa. This would pronounce SloBERTa as the best Slovene model, which is consistent with other tasks and not surprising given that it was trained only on Slovene data.

Analysis of specific tasks shows that none of the models learned anything in MultiRC (all scores are below the Most Frequent baseline). Similar is valid for the BoolQ datasets, where all models but MT-mBERT predict the most frequent class (as explained above, the testing set might be too small for reliable conclusions in BoolQ). We can safely assume that training sample sizes are too small in these two tasks and have to be increased (we have only 92 HT examples in BoolQ and 15 HT examples in MultiRC). To solve that problem, we will train all models on full-size MT datasets in future. Compared to English models, the best two Slovene models achieve good results on WSC and RTE. It seems that none of the English models learns anything from WSC, but the EMBEDDIA SloBERTa model achieves the score of 73.3 (the Most Frequent baseline gives 65.8). Nevertheless, there is still a large gap to human performance. The CroSloEngual model performs better than English BERT on RTE with much fewer training examples (only 9% of the English training data), but lags behind data augmented BERT++. We expected better results on the fully human translated COPA task. We are investigating the reasons for low performance in this task.

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

4.2 Cross-lingual evaluations

The cross-lingual evaluation is split into five subsections according to the type of task. We present results on NER, POS-tagging, dependency parsing, analogies, and SuperGLUE. We train models on a source language dataset in each task and use it for classification in the target language, i.e. we test zero-shot transfer unless specified otherwise. For NER, POS-tagging, and dependency parsing tasks, the results are averaged over five individual evaluation runs, just like in monolingual evaluations. Additionally, ELMoGAN maps were also trained five times and each of the five maps was paired with one of the five classification models for each task during evaluation.

4.2.1 NER

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

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

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

The upper part of the table shows a typical cross-lingual transfer learning scenario, where the model is transferred from resource-rich language (English) to less-resourced languages. In this case, the non-isomorphic ELMoGAN methods, particularly the ELMoGAN-10k variant, are superior to isomorphic mapping with Vecmap and MUSE libraries. In this scenario, ELMoGAN-10k is always the best or close to the best mapping approach. This is not always the case in the lower part of Table 14, which shows the second most important cross-lingual transfer scenario: transfer between similar languages. In this scenario, isomorphic mappings with Vecmap and MUSE are superior. We hypothesise that the reason for the better performance of isomorphic mappings is the similarity of tested language pairs and less violation of the isomorphism assumption the Vecmap and MUSE methods make. The results of the mapping with the MUSE method support this hypothesis. While MUSE performs worst in most cases of transfer from English, the performance gap is smaller for transfer between similar languages. MUSE is sometimes the best method for similar languages, but its results fluctuate considerably between language pairs. The second possible factor explaining the results is the quality of the dictionaries, which are in general better for combinations involving English. In particular, dictionaries obtained by triangulation via English are of poor quality, and non-isomorphic translation might be more affected by imprecise anchor points.

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

In Table 15, we present the results of cross-lingual transfer for contextual BERT models. We compared massively multilingual BERT models (mBERT and XLM-R) with EMBEDDIA trilingual BERT models: Croatian-Slovene-English (CSE), Finnish-Estonian-English (FinEst), and Lithuanian-Latvian-English (LitLat).

Table 15: Comparison of different BERT models, evaluated on the NER task as a zero-shot transfer mode. The best Macro F_1 score for each language pair is in **bold**. The "Reference" column represents a 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.

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

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

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

Finally, the comparison between the best cross-lingual models (in the bottom part of Table 15) and the best monolingual models (reference scores taken from Table 6) shows that with cross-lingual transfer we loose on average 5.2%. (if we excluded the transfer to Lithuanian, which has a problematic dataset, we get even lower gap of 4.4%). This is a very encouraging result, showing that modern cross-lingual technologies have made significant progress and can bridge the technological gap for less-resourced languages. Further, this score is for zero-shot transfer, while a few-shot transfer (with small amounts of data in a target language) might be even closer to monolingual results.

4.2.2 POS-tagging

In Table 16, we present the results of cross-lingual transfer of contextual ELMo embeddings. We compared isomorphic mapping with Vecmap and MUSE libraries and two non-isomorphic mappings using GANs (ELMoGAN-O and ELMoGAN-10k), described in Section 2.2. 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 16: Comparison of different methods for cross-lingual mapping of contextual ELMo embeddings, evaluated on the POS-tagging task. The best micro F_1 score for each language pair is in **bold**. The “Reference“ column represents direct learning on the target language without cross-lingual transfer. The upper part of the table contains a scenario of cross-lingual transfer from English to a less-resourced language, and the lower part of the table shows a transfer between similar languages.

Source.	Target.	Dictionary	Vecmap	ELMoGAN-O	ELMoGAN-10k	MUSE	Reference
English	Croatian	direct	**0.705**	0.629	0.620	0.687	0.963
English	Estonian	direct	0.728	0.678	0.647	**0.729**	0.969
English	Finnish	direct	**0.729**	0.531	0.578	0.715	0.966
English	Latvian	direct	**0.681**	0.625	0.607	0.655	0.940
English	Lithuanian	direct	**0.693**	0.621	0.592	0.640	0.233
English	Russian	direct	0.415	0.488	0.491	**0.665**	0.929
English	Slovenian	direct	0.719	0.637	0.584	**0.723**	0.966
English	Swedish	direct	0.839	0.688	0.649	**0.848**	0.933
Croatian	Slovenian	direct	0.551	0.421	0.435	**0.683**	0.966
Croatian	Slovenian	triang	0.734	0.434	0.461	**0.833**	0.966
Estonian	Finnish	direct	0.586	0.522	0.533	**0.706**	0.966
Estonian	Finnish	triang	0.673	0.514	0.543	**0.690**	0.966
Finnish	Estonian	direct	0.619	0.596	0.590	**0.792**	0.969
Finnish	Estonian	triang	0.703	0.603	0.583	**0.837**	0.969
Latvian	Lithuanian	direct	0.594	0.609	0.591	**0.721**	0.233
Latvian	Lithuanian	triang	0.628	0.627	0.583	**0.724**	0.233
Lithuanian	Latvian	direct	0.238	0.255	0.257	**0.258**	0.940
Lithuanian	Latvian	triang	0.229	**0.257**	0.254	0.256	0.940
Slovenian	Croatian	direct	0.558	0.467	0.495	**0.662**	0.963
Slovenian	Croatian	triang	0.735	0.492	0.502	**0.784**	0.963
Average gap for the best cross-lingual transfer in each language							0.100
Average gap for the best cross-lingual transfer in each language (without Lithuanian)							0.184

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

In Table 17, we present the results of cross-lingual transfer for contextual BERT models. We compared massively multilingual BERT models (mBERT and XLM-R) with EMBEDDIA trilingual BERT models: Croatian-Slovene-English (CSE), Finnish-Estonian-English (FinEst), and Lithuanian-Latvian-English (LitLat).

The results show an advantage of EMBEDDIA 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 (except for Latvian), the average difference being 4.4%.

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

		Non-EMBEDDIA		EMBEDDIA			Best
Source.	Target.	mBERT	XLM-R	CSE	FinEst	LitLat	monolingual
English	Croatian	0.837	**0.846**	0.827	-	-	0.982
English	Estonian	0.799	0.849	-	**0.851**	-	0.977
English	Finnish	0.799	**0.857**	-	0.839	-	0.980
English	Latvian	0.756	0.828	-	-	**0.829**	0.966
English	Lithuanian	0.797	**0.830**	-	-	0.819	0.855
English	Russian	0.812	**0.842**	-	-	-	0.976
English	Slovenian	0.807	**0.834**	0.819	-	-	0.991
English	Swedish	0.908	**0.925**	-	-	-	0.981
Slovenian	Croatian	0.900	0.910	**0.921**	-	-	0.982
Finnish	Estonian	0.834	0.887	-	**0.898**	-	0.977
Estonian	Finnish	0.813	0.889	-	**0.890**	-	0.980
Lithuanian	Latvian	0.795	**0.800**	-	-	0.778	0.966
Latvian	Lithuanian	0.841	**0.878**	-	-	**0.878**	0.855
Croatian	Slovenian	0.895	0.919	**0.924**	-	-	0.991
Average gap for the best cross-lingual transfer in each language							0.075

Similarly to NER, the comparison of ELMo cross-lingual transfer (in Table 16) with variants of multilingual BERT (in Table 17) shows that the transfer with BERT is considerably more successful. The comparison between the best cross-lingual models (these are various BERT models in Table 17) and the best monolingual models (reference scores taken from Table 8) shows that with the cross-lingual transfer we lose on average 7.5%. However, for Lithuanian both XLM-R and LitLat BERT trained on Latvian beat the monolingual reference models.

4.2.3 Dependency parsing

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

The isomorphic mappings with Vecmap are superior in the dependency parsing task, followed by MUSE. Similarly to POS-tagging, the non-isomorphic methods lag. Again, the best cross-lingual ELMo models produce considerably lower scores than the reference model without cross-lingual transfer. The average difference in UAS score is 10.38%, and in LAS it is 24.62% (not considering Lithuanian, which has a failed monolingual ELMo model).

In Table 19, we present the results of cross-lingual transfer for contextual BERT models. We compared massively multilingual BERT models (mBERT and XLM-R) with EMBEDDIA trilingual BERT models: Croatian-Slovene-English (CSE), Finnish-Estonian-English (FinEst), and Lithuanian-Latvian-English (LitLat).

The results show an advantage of EMBEDDIA trilingual models in transfer from English and similar languages (the only difference being the transfer from Lithuanian to Latvian where the XLM-R is more successful, but this dataset is tiny). The transfer from a similar language is more successful than the transfer from English (except for Latvian), the average difference being 11.51%. The comparison between the best BERT cross-lingual models (from Table 19) and the best monolingual models (reference

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

Train	Eval.		Vecmap		ELMoGAN-O		ELMOGAN-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	69.74	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	**67.92**	**39.62**	58.87	28.30	57.36	21.13	66.04	37.74	55.05	24.39
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	**72.38**	**51.43**	60.95	38.10	63.24	36.19	67.62	50.48	55.05	24.39
lv	lt	triang	**75.24**	50.48	62.48	38.48	63.62	36.19	74.29	**53.33**	55.05	24.39
lt	lv	direct	**63.68**	**25.88**	43.50	11.54	50.70	13.69	61.05	18.87	88.85	82.82
lt	lv	triang	**61.86**	**25.94**	49.24	13.31	51.91	13.89	57.95	17.45	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 gap for the best cross-lingual transfer in each language											6.56	17.93
Average gap for the best cross-lingual transfer in each language (without Lithuanian)											10.38	24.62

Table 19: Comparison of different BERT 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.

		Non-EMBEDDIA		EMBEDDIA			Best
Source.	Target.	mBERT	XLM-R	CSE	FinEst	LitLat	monolingual
English	Croatian	42.13	54.00	**56.04**	-	-	82.36
English	Estonian	25.12	38.01	-	**42.30**	-	78.64
English	Finnish	29.08	43.30	-	**46.18**	-	83.64
English	Latvian	23.06	38.66	-	-	**44.93**	74.32
English	Lithuanian	21.32	35.00	-	-	**36.60**	24.34
English	Russian	43.41	**48.19**	-	-	-	80.90
English	Slovenian	38.72	53.90	**58.02**	-	-	85.38
English	Swedish	60.96	**70.79**	-	-	-	80.93
Slovenian	Croatian	52.61	63.66	**67.60**	-	-	82.36
Finnish	Estonian	37.34	53.98	-	**63.08**	-	78.64
Estonian	Finnish	42.11	59.54	-	**67.91**	-	83.64
Lithuanian	Latvian	19.79	**29.98**	-	-	22.35	74.32
Latvian	Lithuanian	27.08	45.38	-	-	**52.83**	24.34
Croatian	Slovenian	52.33	67.16	**71.76**	-	-	85.38
Average gap for the best cross-lingual transfer in each language							12.93
Average gap for the best cross-lingual transfer in each language (without Lithuanian)							18.84

scores taken from Table 10) shows that with the cross-lingual transfer we loose on average 12.93% (not taking Lithuanian with a failed monolingual model into account).

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

4.2.4 Cross-lingual analogies

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

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

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

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

In Table 21, we present the results of contextual BERT models on the cross-lingual analogy task. We compared massively multilingual BERT models (mBERT and XLM-R) with EMBEDDIA 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 from similar languages.

Table 21: Comparison of different BERT 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.

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

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

4.2.5 SuperGLUE tasks

In the cross-lingual scenario, we tested two models (mBERT, CroSloEngual) and transfer between English and Slovene datasets (both directions). For Slovene as the source language, we used the available human translated examples. To make the comparison balanced, we only used the same examples from English datasets. We tested both zero-shot transfer (no training data in the target language) and few-shot transfer. In the few-shot training, we used 10 additional examples from the target language for each task. The fine-tuning hyperparameters are the same as in the monolingual setup.

The results are presented in Table 22. Averaged over all tasks, neither zero-shot nor few-shot learning improves the Most frequent baseline. In general, the models were quite unsuccessful on BoolQ, CB, MultiRC, and WSC but showed promising results on COPA and RTE. 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 COPA and RTE, we can observe that CroSloEngual shows promising results in zero-shot transfer and improves significantly by adding new examples in the target language. Some individual results stand out, e.g. CroSloEngual scoring 70.0 on BoolQ, a significant decrease in performance of mBERT comparing zero-shot and few-shot results on the RTE task, and the 45.4/60.0 score of CroSloEngual on CB. We

Table 22: Cross-lingual results on human translated SuperGLUE test sets. The best results for zero-shot and few-shot scenarios are in bold.

Evaluation	Model	source	target	Avg	BoolQ	CB	COPA	MultiRC	RTE	WSC
					acc.	F1/acc.	Acc.	F1$_a$/EM	Acc.	Acc.
Zero-shot	CroSloEngual	english	slovene	50.0	63.3	23.0/52.7	54.6	53/9.7	62.1	50.7
		slovene	english	**52.2**	63.3	**45.4/60**	**59.0**	**56.3/12.5**	55.2	48.6
	mBERT	english	slovene	51.9	**66.6**	28.4/42.7	50.2	42.4/8.3	**69**	64.4
		slovene	english	46.9	33.3	31.1/55.5	46.2	56.7/11.5	58.6	**65.8**
Few-shot	CroSloEngual	english	slovene	**50.7**	63.3	23.0/52.7	56.4	50.6/11.5	**72.4**	**43.2**
		slovene	english	49.3	**70.0**	23.0/52.7	**62.2**	50.6/11.1	55.2	39.7
	mBERT	english	slovene	43.8	50.0	23.0/52.7	49.8	51.1/8.3	55.2	40.4
		slovene	english	45.6	43.3	23.0/52.7	52.8	**56.9/11.8**	65.5	39.7
	Most frequent			52.4	63.3	23.0/52.7	50.0	77.3/0.3	58.6	65.8

can conclude that for the difficult SuperGLUE benchmark, the cross-lingual transfer is challenging but not impossible.

In the future, we plan to expand the current set of experiments in several directions. First, we will train English models on the full SuperGLUE datasets and transfer them to Slovene human and machine-translated datasets. Second, we will train Slovene models on the combined machine and human translated datasets and transfer them to full English datasets. We will combine Slovene and English training sets and apply the models to both languages. Finally, we will also combine training for several tasks and test transfer learning scenarios.

5 Conclusions and further work

We performed a large scale evaluation of monolingual and cross-lingual embedding approaches developed within WP1 of the EMBEDDIA project. We concentrated on recently most successful contextual embeddings, in particular ELMo and BERT models. For ELMo models, we compared cross-lingual mappings with and without isomorphic assumption. For BERT models, we compared monolingual models, massively multilingual models, and trilingual models. In the evaluation, we used several tasks: NER, POS-tagging, dependency parsing, CoSimLex, analogies, and SuperGLUE benchmarks. We checked the performance of models on nine EMBEDDIA languages: Croatian, English, Estonian, Finnish, Latvian, Lithuanian, Russian, Slovene, and Swedish.

Overall, the results show that EMBEDDIA ELMo models are superior to other ELMo models, but in general, there is a clear advantage of BERT models over ELMo models. In the monolingual setting, monolingual and trilingual BERT models are very competitive, and frequently the EMBEDDIA BERT models dominate. In the cross-lingual setting, BERT models are much more successful compared to ELMo models. The EMBEDDIA trilingual models are mostly better than the massively multilingual models. There are a few exceptions to these general conclusions. The main outlier is the dependency parsing task, where EMBEDDIA ELMo embeddings are better than BERT models.

We can conclude that cross-lingual transfer of trained prediction model is feasible with the presented approaches, especially from similar languages and using specifically designed EMBEDDIA trilingual models. The performance of the best cross-lingual transferred models lags behind the monolingual models for only a few percent, confirming our findings from Deliverable D1.6, described in (Robnik-Šikonja et al., 2021). The exact lag depends on the task and language.

In future work, we will apply the findings to the remaining tasks in WP3, WP4, and WP5 and integrate the developed models into platforms developed in WP6.

6 Associated outputs

The work described in this deliverable has resulted in the following resources:

Description	URL	Availability
ELMo embeddings	`Clarin.si hdl.handle.net/11356/1277`	Public (GPL v3)
CroSloEngual BERT embeddings	`huggingface.co/EMBEDDIA/crosloengual-bert`	Public(CC-BY 4.0)
	`hdl.handle.net/11356/1317`	Public(CC-BY 4.0)
FinEst BERT embeddings	`huggingface.co/EMBEDDIA/finest-bert`	Public(CC-BY 4.0)
	`doi.org/10.15155/9-00-0000-0000-0000-0021CL`	Public(CC-BY 4.0)
LitLatEng BERT embeddings	`huggingface.co/EMBEDDIA/litlat-bert`	Public(CC-BY 4.0)
	`hdl.handle.net/20.500.11821/42`	Public(CC-BY 4.0)
SloBERTa embeddings	`huggingface.co/EMBEDDIA/sloberta`	Public(CC-BY 4.0)
	`hdl.handle.net/11356/1397`	Public(CC-BY 4.0)
Est-RoBERTa embeddings	`huggingface.co/EMBEDDIA/est-roberta`	Public(CC-BY 4.0)
	`doi.org/10.15155/9-00-0000-0000-0000-00226L`	Public(CC-BY 4.0)
Word analogy dataset	`hdl.handle.net/11356/1261`	Public (CC-BY-SA)
Crosslingual NER	`github.com/EMBEDDIA/crosslingual-NER`	Public (GPL v3)
Vecmap changes	`github.com/EMBEDDIA/vecmap-changes`	Public (GPL v3)
ELMoGAN mapping method	`github.com/EMBEDDIA/elmogan`	Public (MIT)
SuPAR ELMo dependency parser	`github.com/EMBEDDIA/super-elmo`	Public (GPL v3)
POS-tagger using ELMo	`github.com/EMBEDDIA/pos-tagging-elmo`	Public (GPL v3)
DP as Sequence Labeling with BERT	`github.com/EMBEDDIA/dep2label-transformers`	Public (MIT)
CoSimLex dataset	`hdl.handle.net/11356/1308`	Public (CC-BY-SA)
Slovene SuperGLUE translation	`hdl.handle.net/11356/1380`	Public (CC-BY-SA)
Slovene SuperGLUE evaluation	`github.com/EMBEDDIA/jiant_slovene`	Public (MIT)

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., . . . 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. *Transactions of the Association for Computational Linguistics*, *5*, 135–146.

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.*

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., . . . 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).

Dozat, T., & Manning, C. D. (2017). Deep biaffine attention for neural dependency parsing. In *Proceedings of 5th international conference on learning representations, ICLR 2017.*

Fu, Z., Xian, Y., Geng, S., Ge, Y., Wang, Y., Dong, X., . . . de Melo, G. (2020). ABSent: Cross-lingual sentence representation mapping with bidirectional GANs. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(05), 7756-7763.

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

Gómez-Rodríguez, C., Strzyz, M., & Vilares, D. (2020). A unifying theory of transition-based and sequence labeling parsing. In *Proceedings of the 28th international conference on computational linguistics* (pp. 3776–3793).

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., … Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672–2680).

Hill, F., Reichart, R., & Korhonen, A. (2015). SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, *41*(4), 665–695.

Hochreiter, S., & Schmidhuber, J. (1997, November). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780.

Jurafsky, D., & Martin, J. H. (2009). *Speech and language processing (2nd edition)*. USA: Prentice-Hall, Inc.

Kittask, C. (2019). *Computational models of concept similarity for the Estonian language* (Bachelor's Thesis). University of Tartu.

Lison, P., & Tiedemann, J. (2016). OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the 10th international conference on language resources and evaluation LREC.*

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., … Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ljubešić, 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.

Martin, L., Muller, B., Ortiz Suárez, P. J., Dupont, Y., Romary, L., de la Clergerie, É., … 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).

Mrkšić, N., Vulić, I., Ó Séaghdha, D., Leviant, I., Reichart, R., Gašić, M., … Young, S. (2017). Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints. *Transactions of the Association for Computational Linguistics*, *5*, 309–324.

Nivre, J., Abrams, M., & Agić, Ž. (2018). *Universal Dependencies 2.3.* Retrieved from `http://hdl.handle.net/11234/1-2895`

Ormazabal, A., Artetxe, M., Labaka, G., Soroa, A., & Agirre, E. (2019). Analyzing the limitations of cross-lingual word embedding mappings. In *Proceedings of the 57th annual meeting of the association for computational linguistics ACL* (pp. 4990–4995).

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).

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

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th annual meeting of the association for computational linguistics: System demonstrations.*

Robnik-Šikonja, M., Reba, K., & Mozetič, I. (2021). Cross-lingual transfer of sentiment classifiers. *Slovenščina 2.0*, *9*(1), 1–25. doi: https://doi.org/10.4312/slo2.0.2021.1.1-25

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. (2020). *MICE: Mining idioms with contextual embeddings.* arXiv preprint 2008.05759. (submitted)

Strzyz, M., Vilares, D., & Gómez-Rodríguez, C. (2019). Viable dependency parsing as sequence labeling. In *Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers)* (pp. 717–723).

Tanvir, H., Kittask, C., & Sirts, K. (2020). *EstBERT: A pretrained language-specific BERT for Estonian.* arXiv preprint 2011.04784.

Ulčar, M., & Robnik-Šikonja, M. (2020). *Cross-lingual alignments of ELMo contextual embeddings.* (draft)

Ulčar, M., & Robnik-Šikonja, M. (2020a). 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., Vaik, K., Lindström, J., Dailidėnaitė, M., & Robnik-Šikonja, M. (2020). Multilingual culture-independent word analogy datasets. In *Proceedings of the 12th Language resources and evaluation conference* (pp. 4067–4073).

Ulčar, M., & Robnik-Šikonja, M. (2020). FinEst BERT and CroSloEngual BERT: less is more in multilingual models. In *Proceedings of Text, Speech, and Dialogue, TSD 2020* (p. 104-111).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).

Venekoski, V., & Vankka, J. (2017). Finnish resources for evaluating language model semantics. In *Proceedings of the 21st Nordic conference on computational linguistics NoDaLiDa* (p. 231-236).

Virtanen, A., Kanerva, J., Ilo, R., Luoma, J., Luotolahti, J., Salakoski, T., ... 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., ... Bowman, S. R. (2019). SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *Advances in Neural Information Processing Systems*, *32*.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., & Bowman, S. (2018). GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP* (pp. 353–355).

Yang, Z., Chen, W., Wang, F., & Xu, B. (2018). Improving neural machine translation with conditional sequence generative adversarial nets. 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. 1346–1355).

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).