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

D1.3: Initial context-dependent and dynamic embeddings technology (T1.2)

Executive summary
The objective of task T1.2 of the EMBEDDIA project is to advance context-dependent and dynamic embeddings technology. In this report we first explain the current scientific context around word embeddings, in which recent developments have produced successful new approaches to context-dependent and dynamic modelling. We describe investigations into applying conventional embedding models to dynamic analysis, and conclude that using the new approaches would be beneficial. We then present currently the most popular contextual embeddings, ELMo and BERT. We analyse their quality depending on the size of training sets and show that existing publicly available ELMo embeddings for EMBEDDIA languages are inadequate. We describe our current work training new ELMo and BERT embeddings on larger training sets, and for the ELMo versions show their advantage over baseline non-contextual FastText embeddings on two benchmarks, the analogy task and the named entity recognition task. To improve objective evaluation of contextual embeddings, we describe the preparation of a novel dataset and task, accepted as a SemEval 2020 challenge.

Partner in charge: QMUL

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:	**D1.3**
Deliverable full title:	**Initial context-dependent and dynamic embeddings technology**
Deliverable short title:	**Initial contextual embeddings**
Document identifier:	**EMBEDDIA-D13-InitialContextualEmbeddings-T12-submitted**
Lead partner short name:	**QMUL**
Report version:	**submitted**
Report submission date:	**31/12/2019**
Dissemination level:	**PU**
Nature:	**R = Report**
Lead author(s):	**Matthew Purver (QMUL), Marko Robnik-Šikonja (UL), Matej Ulčar (UL)**
Co-author(s):	**Carlos Santos Armendariz (QMUL)**
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
19/09/2019	v1.0	Matej Ulčar (UL)	Initial version.
01/10/2019	v1.1	Matthew Purver (QMUL)	The first version.
12/11/2019	v1.2	Marko Robnik-Šikonja (UL)	Check, rewrite, and consolidate.
12/11/2019	v1.3	Matej Ulčar (UL)	Added evaluation.
18/11/2019	v1.4	Marko Robnik-Šikonja (UL)	Check, rewrite, and consolidate.
20/11/2019	v1.5	Matthew Purver (QMUL)	Updated template, added outputs section.
21/11/2019	v1.6	Matthew Purver, Carlos Santos Armendariz (QMUL)	Updated GWSC section.
28/11/2019	v1.7	Mark Granroth Wilding (UH)	Internal review.
06/12/2019	v1.8	Matthew Purver (QMUL), Matej Ulcar, Marko Robnik-Sikonja (UL)	Reponses to internal review.
10/12/2019	v1.9	Nada Lavrač (JSI)	Report quality checked.
18/12/2019	v1.10	Matthew Purver (QMUL)	Revision after quality control.
20/12/2019	final	Senja Pollak (JSI)	Report quality checked and finalised.
23/12/2019	submitted	Tina Anžič (JSI)	Report submitted.

Table of Contents

List of abbreviations

biLM	Bi-directional Language Model
NLP	Natural Language Processing
NER	Named Entity Recognition
LSTM	Long Short-term Memory
CSLS	Cross-domain Similarity Local Scaling
CNN	Convolutional Neural Network
ELMo	Embeddings from Language Models
BERT	Bidirectional Encoder Representations from Transformers
CBOW	Continuous Bag Of Words
MLM	Masked Language Model

1 Introduction

The EMBEDDIA project aims to improve cross-lingual transfer of language resources and trained models using word embeddings and cross-lingual word embeddings. We presented the basic description of embeddings and cross-lingual embeddings in *D1.1 Datasets, benchmarks and evaluation metrics for cross-lingual word embeddings*. To make this document self-contained, we first repeat some basic explanations in Section 1.1, which a reader acquainted with embeddings can skip. Section 1.2 outlines the context of this deliverable within the EMBEDDIA project and presents the structure of this report.

1.1 Introducing embeddings

To process text, neural networks require numerical representation of the given text (words, sentences, documents), referred to as **text embeddings**. In this work we focus on **word embeddings**, which are representations of words in numerical form, consisting of vectors of typically several hundred dimensions. The vectors are used as an input to machine learning models; for complex language processing tasks these are typically deep neural networks. These embedding vectors are learned from large monolingual text collections (called corpora), originally by direct statistical inference: by characterising words in terms of the words with which they co-occur, the representations exploit the *distributional hypothesis* that word meaning is reflected in its context of use (Firth, 1957). Alternatively, and more commonly in the recent work, embedding vectors can be derived using specialized learning tasks based on neural networks, e.g., word2vec (Mikolov, Le, & Sutskever, 2013), GloVe (Pennington et al., 2014), or FastText (Bojanowski et al., 2017). In either case, the resulting embeddings encode important information about word meaning as distances between vectors, and capture semantic relations between words.

Probably the best known word embedding approach is the word2vec method (Mikolov, Sutskever, et al., 2013) which we use as a baseline. The problem with word2vec embeddings is their failure to express polysemous words. During training of an embedding, all senses of a given word (e.g., *paper* as a material, as a newspaper, as a scientific work, and as an exam) contribute relevant information in proportion to their frequency in the training corpus. This causes the final vector to be placed somewhere in the weighted middle of all words' meanings. Consequently, rare meanings of words are poorly expressed with word2vec and the resulting vectors do not offer good semantic representations of these words. For example, none of the 50 closest vectors of the word *paper* is related to science.[1]

The idea of **contextual embeddings** is to generate a different vector for each context a word appears in, with this context typically being defined sentence-wise. To a large extent, this solves the problems with word polysemy: the context of a sentence is typically enough to disambiguate different meanings of a word for humans, and so it is for the learning algorithms. It has also been shown to provide improvements in performance across a range of NLP tasks, by better reflecting the relationship between a word and its context of use (see e.g. Peters et al., 2018; Devlin et al., 2019).

Modern word embedding spaces exhibit similar structures across languages, even when considering distant language pairs like English and Vietnamese (Mikolov, Le, & Sutskever, 2013). This means that embeddings independently produced from monolingual text resources can be aligned (Mikolov, Le, & Sutskever, 2013), resulting in a common cross-lingual representation, called **cross-lingual embedding**, which allows for fast and effective integration of information in different languages.

As explained in deliverable D1.2, the state-of-the-art in embeddings is rapidly progressing (for example, at the time the EMBEDDIA project was conceived, the methods for training contextual embeddings like ELMo and BERT did not exist). In this work, we mostly use, analyse, and improve upon the currently most widely used ELMo (Peters et al., 2018) approach to contextual word embeddings, while the experiments using the BERT (Devlin et al., 2019) approach are still ongoing.

[1] A demo showing near vectors computed with word2vec from Google News corpus is available at `http://bionlp-www.utu.fi/wv_demo/`.

1.2 Objectives and structure of the report

The objectives of workpackage WP1 of the EMBEDDIA project are to advance cross-lingual and context-dependent word embeddings and test them with deep neural networks in the context of nine European languages: English, Slovene, Croatian, Estonian, Lithuanian, Latvian, Russian, Finnish, and Swedish. This report describes the results of the work performed in T1.2 in the first 12 months of project duration. The specific objective of T1.2 is to advance context-dependent and dynamic embeddings technology. The main contributions of this work are listed below in the order of appearance.

- an experiment into the use of dynamic techniques with standard static embedding models, showing the need for more robust approaches, in Section 2;
- an analysis, showing the need for language specific improved ELMo embeddings, trained on larger corpora, presented in Section 3.1.1;
- evaluation of these newly produced ELMo embeddings on word analogy and NER datasets, presented in Section 4 and in the appended paper by Ulčar & Robnik-Šikonja (2019), submitted to the LREC-2020 conference;
- a newly designed evaluation task named Graded Word Similarity in Context (GWSC), with a novel accompanying dataset for contextual embeddings (CoSimLex); the task has been accepted as a SemEval 2020 shared task and the dataset will be made publicly available thereafter. These are presented in Section 5 and in the appended paper by Santos Armendariz et al. (2019), submitted to the LREC-2020 conference.

The above objectives and contributions are slightly different from the ones anticipated in the EMBEDDIA project proposal: at the time of proposal writing we anticipated that context-dependent and dynamic methods would need to be developed from scratch. With *context-dependent* modelling, our intention was to model the fact that word meaning depends on the particular context in which a word token appears: every usage of a word takes place in some sentential, lexical or discourse context and this has an effect on the meaning that a reader or hearer takes it to have. A context-dependent model should therefore produce a different vector representation for every occurrence of a word in a text. The intention behind *dynamic* models, on the other hand, was the ability to model the longer-term, more general changes in word meaning that happen over time or between domains and genres. A dynamic model should therefore produce a different vector for a word type depending on the time period or domain in which it occurs, but not necessarily for every word token occurrence within that period/domain. The two concepts are not mutually exclusive: an ideal embedding model might be both context-dependent and dynamic.

Indeed, recent advances in NLP have developed highly successful embedding approaches which are both context-dependent and dynamic according to these definitions, for example ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019). This means that part of the original EMBEDDIA objectives have already been successfully achieved by the AllenNLP (ELMo) and Google (BERT) research groups, with huge amounts of available resources; these models have also been shown to give very good performance, advancing the state of the art across many NLP tasks, due to their combination of context-dependence and the ability to incorporate large amounts of information via transfer learning and pre-training. Given this, ELMo, BERT and related models address some of the objectives of Task T1.2 of EMBEDDIA, and we have therefore in T1.2 partially re-focused our research on these models and describe them in this deliverable together with our own modifications and advances.

The work reported in this deliverable (stemming from Task T1.2) is closely related to the work done in Tasks T1.1 and T1.3, described in deliverables D1.2 (cross-lingual embeddings) and D1.4 (deep neural network architectures). Together these tasks and deliverables describe the core embedding technologies as a prerequisite for successful application of deep neural networks in (cross-lingual) text processing.

This report is split into five further sections. Section 2 describes initial investigations into achieving dynamic effects with traditional static, non-context-dependent models, showing the need for the im-

provements given by state-of-the-art contextual models. In Section 3 we describe the architecture and training methods of contextual embedding models, as well as the parameters and datasets used in the development of our new ELMo vectors and BERT models. The evaluation methods and the initial evaluation results for the produced embeddings are presented in Section 4. We propose a novel approach to test the contextual embeddings in Section 5. We list the software and models developed and released as outputs in Section 6, and present conclusions about the context-dependent embeddings in Section 7 where we also outline the plans for further work. The two appendices then include the papers by Ulčar & Robnik-Šikonja (2019) and Santos Armendariz et al. (2019).

2 Dynamic embeddings for diachronic changes

We first describe our initial research in *dynamic* embeddings, which was based on conventional, non-context-dependent ("static") embeddings as produced by models such as word2vec (Mikolov, Sutskever, et al., 2013) or (in this case) GloVE (Pennington et al., 2014). Our intention was to investigate diachronic changes in meaning (as reflected in usage) over a relatively short period of about 10 years. This might be needed when analysing effects in news stories covering extended topics over time. Following on from our own earlier work (Purver et al., 2018), we applied this to the domain of financial reporting, using a corpus of company financial report documents rather than news media, and investigating changes related to the financial crisis of 2007-8. Although the text domain is not completely representative of the EMBEDDIA use cases, the applicability of the techniques is transferable.

We followed the method of Hamilton et al. (2016), in which conventional static GloVE models (Pennington et al., 2014) are learned separately on datasets for specific time periods, and aligned with each other using a linear transformation learned by the Orthogonal Procrustes technique. This technique has been shown very effective at revealing long-term changes in word meaning in cases where the changes are relatively large compared to the vocabulary in general; for example, the shift in meaning over centuries of words such as *gay* (from 'light-hearted, carefree' to 'homosexual') and *broadcast* (from 'scattering of seed' to 'transmission of media information') (see Hamilton et al., 2016). In our case, however, the changes in question are more subtle and take place over a shorter time.

Here, we show results for two terms of interest investigated: *impairment* and *dividend*. Asset impairment is an accounting procedure by which firms can recognise past investment mistakes; its use is highly discretionary and can be exploited as a way of reporting losses while maintaining market credibility (Riedl, 2004). Dividends to shareholders can be a strong signal of company financial health, with their use varying with market conditions. Both terms might therefore be expected to be used in varying ways before, during and after a period of financial crisis.

Figures 1 and 2 show the changes over time in the similarity between the term *impairment* and its closest neighbours, visualised in two dimensions using t-SNE (Van der Maaten & Hinton, 2008); Figure 3 then shows the same for the term *dividend*. Each word is shown as a line connecting the points representing its embedding vectors for the years 2003-2012, with the arrowhead at the final year to indicate direction. The two central years of the crisis, 2007-8, are shown as solid blocks. The t-SNE visualisations allow us to get an impression of the similarity and/or association between words, and how these change. However, t-SNE does not accurately preserve distances between all points (rather, it tries to represent clusters and the distances between them, but at the expense of representing intra- and inter-cluster distances differently). These visualisations can therefore allow us to find apparent patterns and changes. Assessing their strength and confirming their direction must be done separately.

Some interesting changes are visible: for example, the association between *impairment* and *goodwill* (a type of intangible asset associated with company purchase) does seem to be affected by the crisis, with similarity decreasing to a low during the years 2006-8; the association between *impairment* and *testing* follows a general increasing trend over time although with a dip during the same crisis years. *Dividend* becomes less similar to *payment* over time, suggesting potential for conclusions about the use of dividend payments.

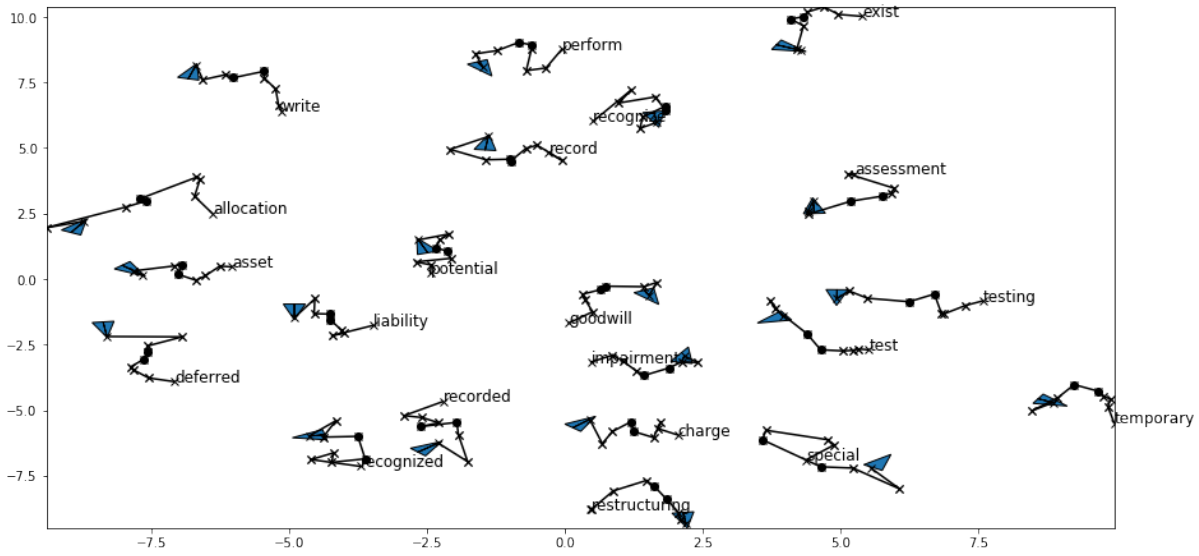

Figure 1: Changes in associations over time in the space centered on *impairment*, as visualised by t-SNE.

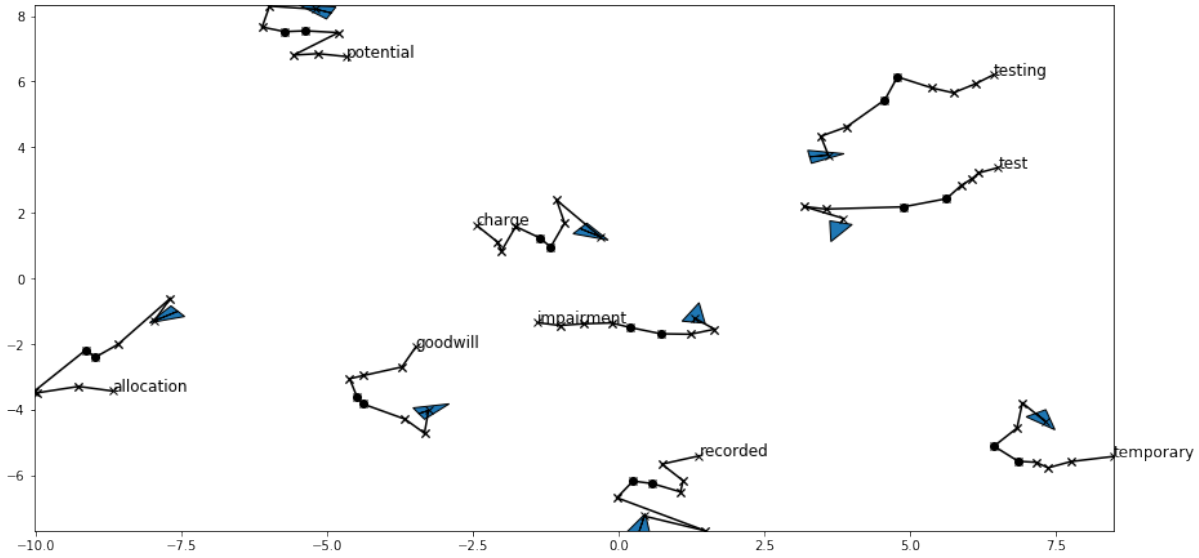

Figure 2: Changes in associations over time in the space centered on *impairment*, as visualised by t-SNE; zoomed in to only 5 nearest neighbours (per year).

However, this relationship seems to differ with different morphological variants (*pay*, *paid*). Other neighbours that might be expected to be informative seem to show little change. Relative to *impairment*, for example, the closest neighbour *charge* moves in a similar direction with no clear change in separation between the two. For this domain and use case, this method seems to have some significant limitations. Primarily, the need to use relatively small datasets for individual time periods to build independent embedding spaces — which leads to changing representations for the entire vocabulary — makes analysis of specific changes hard. This is further complicated by the fact that the static embeddings models on which the dynamic diachronic analysis is based do not account for individual word context of occurrence, forcing all uses to be represented as one point and effectively introducing further noise into the representations. Our focus has therefore now shifted to using the new state-of-the-art contextual models such as ELMo and BERT, which solve the latter problem directly, while also addressing the former

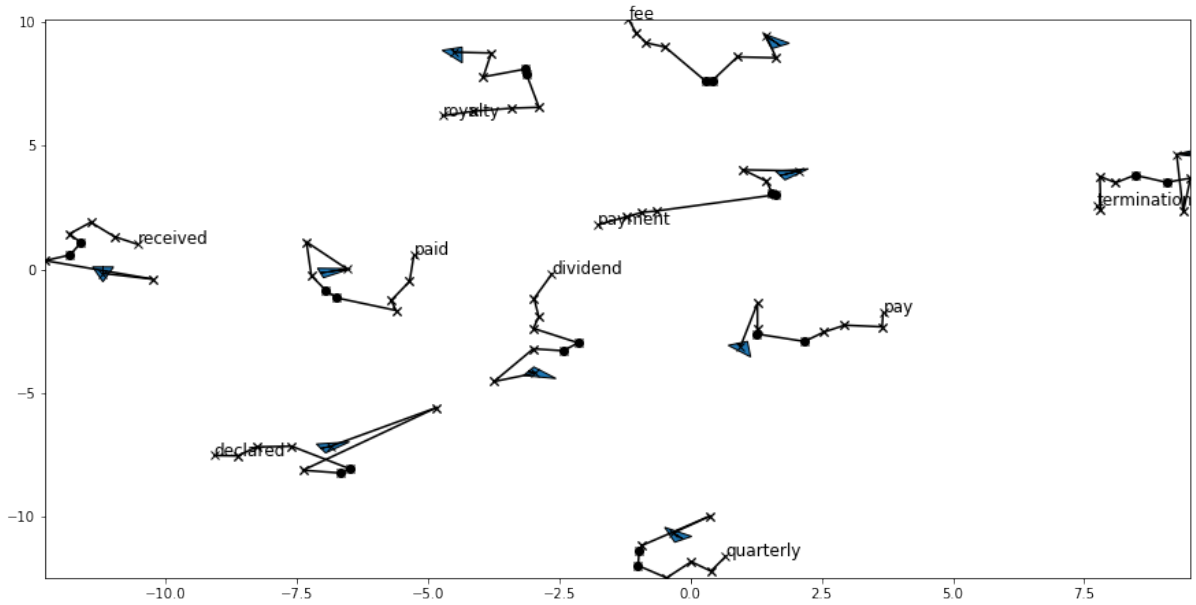

Figure 3: Changes in associations over time in the space centered on *dividend*, as visualised by t-SNE; zoomed in to only 5 nearest neighbours (per year).

problem via transfer learning—allowing baseline models to be learned from very large datasets and then fine-tuned to specific small datasets. The following sections describe our work on this kind of models. We will report on their application to dynamic changes in future deliverables, including work currently under way on a BERT-based model for diachronic analysis.

3 Existing and new context-dependent embeddings

Context-dependent embeddings are generated by language models that take into account the context, and generate word embedding vectors for each occurrence of a word, not just for each given word: the same word in different contexts will be assigned different embeddings. In this section, we describe ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019) contextual models. These models can be trained on very large text datasets to produce high quality contextual embeddings, and these can be used directly in a range of NLP tasks, or fine-tuned for them, in a form of transfer learning. We describe existing pre-computed models and the new ones we have generated for all EMBEDDIA languages.

3.1 ELMo

ELMo (Embeddings from Language Models) (Peters et al., 2018) is one of the state-of-the-art pretrained transfer learning models. The ELMo model's architecture consists of three neural network layers. The output of the model after each layer gives one set of embeddings, altogether three sets. The first network layer is convolutional (CNN) and operates on a character level. It is context independent, so each word always gets the same embedding, regardless of its context. This layer is followed by two biLM layers. A biLM layer consists of two concatenated LSTMs. The first, left-to-right LSTM is trained to predict the following word, based on the given past words, where each word is represented by the embeddings from the CNN layer. The second, right-to-left LSTM tries to predict the preceding word, based on the given following words. Although ELMo is trained on character-level input and is able to handle out-of-vocabulary words, a vocabulary file containing the most common tokens is used for efficiency during training and embedding generation.

In NLP tasks, usually a weighted average of these embeddings is used. The weights for merging the representation of layers are learned during the training of the model for a specific task. Additionally, the entire ELMo model can be fine-tuned for the specific task.

3.1.1 Existing ELMo embeddings

The original ELMo model (Peters et al., 2018) was trained on a 1 billion word English corpus with a vocabulary file of about 800,000 words (see Table 1). Later, ELMo models for other larger languages were trained, like Chinese and Spanish. Recently, ELMoForManyLangs (Che et al., 2018) project released pre-trained ELMo models for many languages (Fares et al., 2017). These models, however, were trained on significantly smaller datasets. They used 20 million words datasets randomly sampled from the raw text released by the CONLL 2017 shared task (wikidump + common crawl). The quality of these models is questionable. For example, we compared the Latvian model from ELMoForManyLangs with a model we trained on the complete CONLL 2017 Latvian corpus (wikidump + common crawl) with about 280 million tokens. The difference between models on all the categories of the word analogy task are shown in Figure 4. As the results of the ELMoForManyLangs embeddings are significantly worse than using the full corpus, we can conclude that these embeddings are not of sufficient quality. For this reason, we decided not to use ELMoForManyLangs models, but to train our own models instead.

An open source DeepPavlov library contains three different pre-trained ELMo models for Russian.[2] We tested the model trained on the Russian WMT News dataset, which contains 946 million tokens. Comparison of the Russian ELMo model by DeepPavlov with some of our ELMo models on the word analogy task is shown in Figure 5 in Section 4. As the Russian model by DeepPavlov performs similarly to our ELMo models for other languages, we decided to use this model rather than train our own Russian ELMo model.

3.1.2 Newly developed ELMo embeddings

To obtain contextual embeddings of sufficient quality for all EMBEDDIA languages except English and Russian, we generated ELMo embeddings on the monolingual corpora we obtained in T1.5 (see deliverable *D1.1 Datasets, benchmarks and evaluation metrics for cross-lingual word embeddings*). For each language, we used a corpus with several hundred million tokens. All corpora were sentence and word tokenized before training. We used the NLTK tokenizer[3] for corpora that were not already pre-tokenized. For each language we prepared a vocabulary file, containing roughly one million most common tokens: words, numbers and punctuation marks that appear at least n times in the corpus, where n is between 15 and 25, depending on the corpus size. We trained all ELMo models for about 10 epochs. Details about datasets used for training ELMo models are presented in Section 3.3 and in the paper of Appendix A. The resulting models have been deposited to the CLARIN repository and are publicly available (see Section 6).

3.2 BERT

The architecture of the BERT model (Devlin et al., 2019) is composed of 12 layers of Transformer encoder cells (Vaswani et al., 2017). BERT uses a masked language model (MLM) to randomly hide a given percentage of tokens and predict them during training. A final standard neural network classifier layer is then trained to produce an output, standardly to predict whether two given sentences are in consecutive order or not (although it can be trained to perform other tasks).

Unlike ELMo, BERT is not trained with a character level input, but uses subword tokens. Some very

[2] https://github.com/deepmipt/DeepPavlov
[3] https://www.nltk.org/

common words are kept as single tokens, others are split into common stems, prefixes, etc. and sometimes down to single-letter tokens.

3.2.1 Existing BERT models

The original BERT project offers pre-trained English, Chinese, Spanish, and multilingual models. The multilingual BERT model is trained simultaneously on 104 languages, including all EMBEDDIA languages, using very large amounts of data; it therefore provides a model in which the languages are embedded in the same space, without requiring further explicit cross-lingual mapping; but may be suboptimal for any specific language or subset of languages.

An open source DeepPavlov library[4] offers a specific Russian BERT model, just as for ELMo.

3.2.2 New BERT models under development

Given that there are no language-specific BERT models for EMBEDDIA languages other than the English and Russian versions mentioned above, we decided to train new BERT models. To this end we used bert-vocab-builder[5] to produce subword vocabulary from a given corpus. We randomly mask 15% of the tokens in the corpus and repeat the process 5 times, each time with different 15% of the tokens being masked. We have built such training data for Slovene-Croatian-English and Estonian-Finnish-English multilingual BERT models. The corpora used to train our BERT models will be the same as the corpora used to train ELMo models and are described in Section 3.3. At the time of writing this report, we have not yet completed training of these models (training requires a long time with significant computational resources). The produced models will be deposited to the CLARIN repository.

3.3 Training corpora

We train ELMo and BERT models for all the EMBEDDIA languages on monolingual corpora from various sources. Some are available online under permissive licences, some are available only for research, and some were provided by the project partners. The details of the size of corpora used for each language are displayed in Table 1. Further details can be found in Deliverable D1.1 *Datasets, benchmarks and evaluation metrics for cross-lingual word embeddings*.

Table 1: The training corpora for ELMo and BERT embeddings and their properties: size (in billions of tokens) and ELMo vocabulary size (in millions of tokens).

Language	Corpora	Size	Vocabulary
Croatian	hrWaC 2.1, Riznica, Styria articles	1.95	1.4
English	1 Billion Word Benchmark	0.8	0.8
Estonian	CoNLL 2017, Ekspress Meedia articles	0.68	1.2
Finnish	STT articles, CoNLL 2017, Ylilauta downloadable version	0.92	1.3
Latvian	CoNLL 2017	0.27	0.6
Lithuanian	Wikipedia 2018, DGT-UD	0.12	0.4
Russian	WMT News	0.95	1.0
Slovene	Gigafida 2.0	1.26	1.4
Swedish	CoNLL 2017, STT articles	1.68	1.2

[4] https://github.com/deepmipt/DeepPavlov
[5] https://github.com/kwonmha/bert-vocab-builder

4 Initial evaluation

We evaluated the produced ELMo models for all EMBEDDIA languages and BERT models computed till the time of this writing. We used two evaluation tasks: the word analogy task and named entity recognition (NER) task. Details of these tasks are described in Deliverable D1.1 *Datasets, benchmarks and evaluation metrics for cross-lingual word embeddings*.

4.1 Analogy task

The word analogy dataset contains 15 different categories, 5 semantic and 10 syntactic/morphologic. Each instance of analogy contains only four words, without any context, so the contextual models do not have enough context to generate sensible embeddings. We therefore used some additional text to form simple sentences using the four analogy words, while taking care that their noun case stays the same. For example, for the words "Rome", "Italy", "Paris" and "France" (forming the analogy Rome is to Italy as Paris is to x, where the correct answer is x =France), we formed the sentence "If the word Rome corresponds to the word Italy, then the word Paris corresponds to the word France". We generated embeddings for those four words in the constructed sentence, substituted the last word with each word in our vocabulary and generated the embeddings again. As typical for non-contextual analogy task, we measure the cosine distance (d) between the last word (w_4) and the combination of the first three words ($w_2 - w_1 + w_3$). We use the CSLS metric (Conneau et al., 2018) to find the closest candidate word (w_4). If we find the correct word among the five closest words, we consider that entry as successfully identified. The proportion of correctly identified words forms a statistics called accuracy@5, which we report as the result.

As the training of our BERT models is not yet complete (see above), we focus on ELMo here. We first compare existing ELMo embeddings from ELMoForManyLangs project with our embeddings, followed by the detailed analysis of our ELMo embeddings.

4.1.1 Existing ELMo embeddings

In the first evaluation, we tested the existing ELMo embeddings from the ELMoForManyLangs project, and compared them with our new versions. Recent developments in pretrained models (Devlin et al., 2019; Conneau et al., 2019) have shown that the size of the training corpus is a crucial factor in the quality of produced embeddings. As a confirming example, we show here the comparison of the existing Latvian language ELMo embeddings with our ELMo embeddings, build on a much larger corpus. We used the embedding vectors from the first biLM (LSTM) layer for comparison, because embeddings from this layer give the best results on this task, especially on semantic categories (Table 2). The results for each category in both Latvian ELMo models and EMBEDDIA Estonian ELMo model are presented in Figure 4.

The Latvian ELMo model from ELMoForManyLangs project performs significantly worse than the EMBEDDIA ELMo Latvian model on all categories of the word analogy task. We include comparison with our Estonian ELMo embeddings in the same figure. This comparison shows that while differences between our Latvian and Estonian embedding can be significant for certain categories, the accuracy score of ELMoForManyManyLangs is always worse than either of our models.

In Figure 5, we compare the Russian ELMo model from the DeepPavlov library with the EMBEDDIA ELMo models for Estonian, Croatian, and Slovene. The Russian ELMo model does not stand out from the other models, neither due to positive nor negative performance. Although the difference in corpus sizes between Estonian and Croatian is nearly three-fold (Table 1), the difference in performance on the word analogy task is minor; in fact, the Estonian model outperforms the Croatian model in several categories. The comparison of Estonian and Latvian models leads us to believe that a few hundred

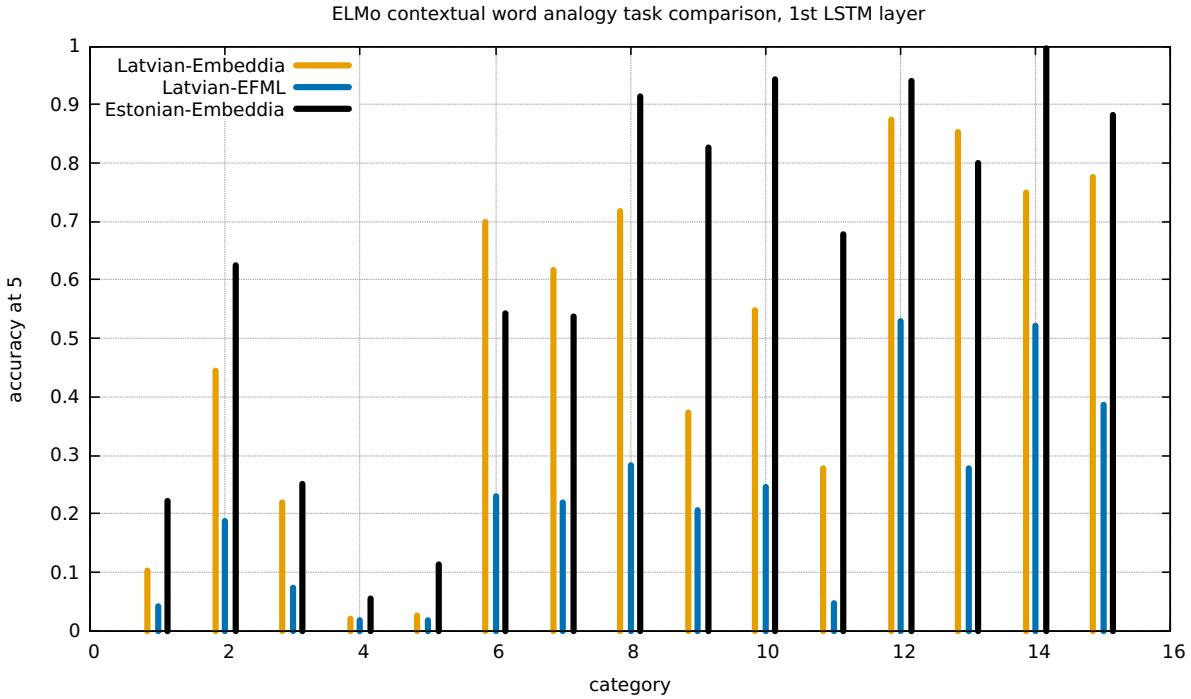

Figure 4: Comparison of Latvian ELMo model by ELMoForManyLangs (blue, latvian-efml), Latvian ELMo model trained by EMBEDDIA team (yellow, latvian-embeddia), and Estonian ELMo model trained by EMBEDDIA team (black, estonian-embeddia). The performance is measured as accuracy@5 on the word analogy task, where categories 1 to 5 are semantic, and categories 6 to 15 are syntactic. The embeddings use weights of the first biLM/LSTM layer (i.e. the second layer overall).

million tokens is a sufficiently large corpus to train ELMo models (at least for the word analogy task), but the 20-million token corpora used in ELMoForManyLangs are too small.

4.1.2 EMBEDDIA ELMo embeddings

The results for all EMBEDDIA languages and embeddings formed from all ELMo layers, averaged over semantic and syntactic categories of analogy task, are shown in Table 2. The embeddings after the first LSTM layer perform the best on the semantic categories. On the syntactic categories, the non-contextual CNN layer (layer 0) performs the best. Syntactic categories are less context dependent and much more morphology and syntax based, so it is not surprising that the non-contextual layer performs well. The second LSTM layer embeddings perform the worst on the syntactic categories, though still outperforming CNN layer embeddings on the semantic categories. The Latvian ELMo model performs worse compared to other languages we trained, especially on the semantic categories, presumably due to smaller training data size. Surprisingly, the original English ELMo model performs poorly on the syntactic categories and only outperforms Latvian on the semantic categories. The low score can be partially explained by English model scoring 0.00 for one syntactic category “opposite adjective”, which we are currently not able to explain.

4.2 NER task

As described in Deliverable D1.1 *Datasets, benchmarks and evaluation metrics for cross-lingual word embeddings*, the labels we used in the NER datasets are simplified to a common set of three labels (person - PER, location - LOC, organization - ORG), present in all the EMBEDDIA languages. Each word in the

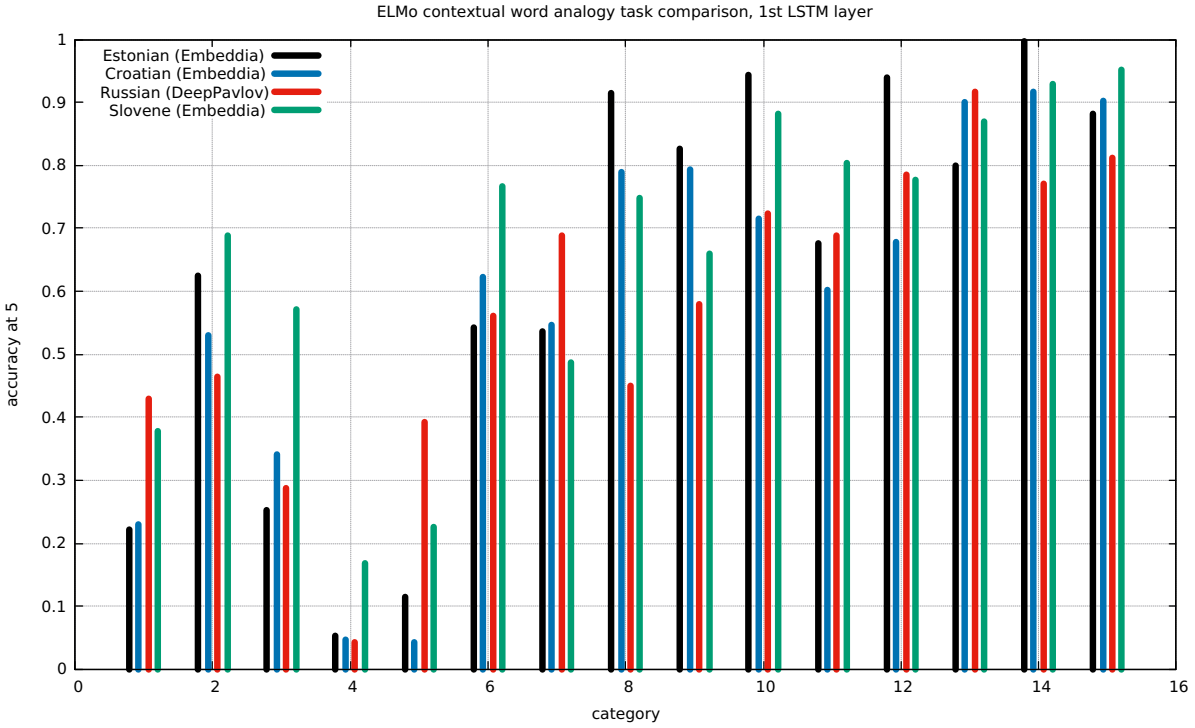

Figure 5: Comparison of Estonian (black), Croatian (blue), Slovene (green) EMBEDDIA ELMo models, and Russian ELMo model by DeepPavlov (red) on word analogy task. As embeddings we use neural network weights after the 1st biLM/LSTM layer. In the word analogy task, the categories 1 to 5 are semantic, and the categories 6 to 15 are syntactic.

Table 2: The results of word analogy evaluation task, using accuracy@5 score for each language (shown as 2-letter ISO codes). Results are shown for embeddings composed separately from each layer of ELMo network and are averaged over all semantic and all syntactic categories. The best results for each language and category are emphasised with the bold typeface.

category	semantic			syntactic		
layer	CNN	LSTM1	LSTM2	CNN	LSTM1	LSTM2
en	0.18	**0.21**	**0.21**	**0.22**	**0.22**	0.21
hr	0.13	**0.24**	0.20	**0.79**	0.75	0.54
et	0.10	**0.25**	0.18	**0.85**	0.81	0.63
fi	0.13	**0.33**	0.25	**0.83**	0.74	0.54
lv	0.08	**0.16**	0.13	**0.74**	0.65	0.43
lt	0.08	**0.29**	0.21	**0.86**	**0.86**	0.62
sl	0.14	**0.41**	0.33	**0.79**	**0.79**	0.57
sv	0.21	**0.25**	0.22	**0.80**	0.60	0.34
ru	0.13	**0.32**	0.19	**0.77**	0.70	0.53

NER dataset is labelled with one of the three labels or the label 'O' (i.e., Other, if it does not fit any of the other three labels). The frequency of these labels in the datasets for each language is shown in Table 3.

To measure the performance of ELMo embeddings on the NER task we proceeded as follows. We embedded the text in the datasets sentence by sentence, producing three vectors (one from each ELMo layer) for each token in a sentence. Reimers & Gurevych (2019) show that for the NER task fixed averaging of the layers, weighted averaging of the layers and concatenating layers all work well. Although

Table 3: The number of words labelled with each of the named entity labels (PER, LOC, ORG) and the density of these labels (their sum divided by number of all words) for datasets in all EMBEDDIA languages.

Language	PER	LOC	ORG	density
Croatian	3931	656	1138	0.064
Estonian	8490	6326	6149	0.096
Finnish	3402	2173	11258	0.087
Latvian	5615	2643	3341	0.085
Lithuanian	2101	2757	2126	0.076
Slovenian	4478	2460	2667	0.049
Swedish	3976	1797	1519	0.047
English	17050	12316	14613	0.146
Russian	3293	2738	3635	0.107

fixed average is slightly worse than the other two approaches, we decided to use it to keep the memory requirements low and the speed of the training high. We calculated the average of the three vectors and used it as the input of our recognition model. The input layer was followed by a single LSTM layer with 128 LSTM cells and a dropout layer, randomly dropping 10% of the neurons on both the output and the recurrent branch. The final layer of our model was a time-distributed softmax layer with 4 neurons.

We used the ADAM optimiser (Kingma & Ba, 2014) with learning rate 0.01 and 10^{-5} learning rate decay. We used categorical cross-entropy as a loss function and trained the model for 3 epochs. We present results using the Macro F_1 score, that is the average of F_1-scores for each of the three NE classes (the class Other is excluded).

Since the differences between the tested languages depend more on the properties of the NER datasets than on the quality of embeddings, we can not directly compare ELMo models. For this reason, we take the non-contextual FastText embeddings[6] as a baseline and predict named entities using them. The architecture of the model using FastText embeddings is the same as the one using ELMo embeddings, except that the input uses 300 dimensional FastText embedding vectors, and the model was trained for 5 epochs (instead of 3 as for ELMo). In both cases (ELMo and FastText) we trained and evaluated the model five times, because there is some random component involved in initialization of the neural network model. By training and evaluating multiple times, we minimise this random component.

The results are presented in Table 4. We included the evaluation of the original ELMo English model in the same table. NER models have little difficulty distinguishing between types of named entities, but recognizing whether a word is a named entity or not is more difficult. For languages with the smallest NER datasets, Croatian and Lithuanian, ELMo embeddings show the largest improvement over FastText embeddings. However, we can observe significant improvements with ELMo also on English and Finnish, which are among the largest datasets (English being by far the largest). Only on Slovenian dataset did ELMo perform slightly worse than FastText, on all other EMBEDDIA languages, the ELMo embeddings improve the results.

5 Graded Word Similarity in Context

Most intrinsic evaluation methods for embeddings do not take context into account, but are based only on properties of words when seen in isolation; for example, on the ability of an embedding model to predict human judgements of similarity between pairs of words as recorded in resources like SimLex (Hill et al., 2015). Some recent work has introduced context-dependence into this kind of intrinsic evaluation, by measuring similarity between uses in different sentential contexts (Huang et al., 2012; Pilehvar & Camacho-Collados, 2018). However, so far this has assumed that the object of study for evaluation purposes is words with distinct discrete meanings (*polysemous* words); as such, it is not

[6] https://fasttext.cc/

Table 4: The results of NER evaluation task, averaged over 5 training and evaluation runs. The scores are average F_1 score of 3 named entity classes. The columns show FastText, ELMo, and the difference between them ($\Delta(E - FT)$).

Language	FastText	ELMo	$\Delta(E - FT)$
Croatian	0.17	0.53	0.36
Estonian	0.26	0.31	0.05
Finnish	0.71	0.84	0.13
Latvian	0.39	0.45	0.06
Lithuanian	0.43	0.65	0.22
Slovenian	0.68	0.67	-0.01
Swedish	0.82	0.88	0.06
English	0.28	0.43	0.15
Russian	0.53	0.56	0.03

fully suitable for evaluation of embedding models that assign different representations to words in all contexts, or the ability of these models to reflect the subtle, graded changes in meaning that humans perceive. We have therefore developed a new evaluation task, designed to solve these problems and allow a full intrinsic evaluation of context-dependent embeddings in terms of word similarity. This task has been accepted as part of the 2020 edition of the SemEval challenge: SemEval (the International Workshop on Semantic Evaluation) is an annual series of public challenges in the evaluation of systems for computational semantics.[7] Our task, named *Graded Word Similarity in Context (GWSC)*,[8] will run as a public competition in Feburary 2020;[9] the multilingual dataset we are creating will be distributed to participants and publicly released once the competition is over.

5.1 Task description

The goal of GWSC is to predict graded word similarity in context, for multi-lingual data. Systems entered for the task will be presented with a paragraph of text, and must predict human judgements of the similarity of meaning of two words as they appear within that context. Each pair of words will be presented in two different contexts, and thus paired with two corresponding different gold standard judgements (see Figure 6); contexts will be chosen so as to encourage different perceptions of similarity, and models must therefore be context-aware in order to perform well on the task. The task is multi-lingual, with datasets provided in at least four EMBEDDIA languages (English, Slovenian, Croatian, Finnish; we plan to add Estonian if time allows). The examples are not restricted to polysemous words but include examples of more subtle, graded changes in meaning.

The SemEval GWSC task is to be unsupervised: systems will not be given training data, as the intention is to evaluate general embedding spaces, rather than classifiers trained only for this specific task. We will evaluate performance on two subtasks and a baseline:

Predicting Ratings: participating systems must predict the absolute similarity rating for each word pair in each context. This will be evaluated using Spearman correlation with gold-standard judgements, following the standard evaluation methodology for similarity datasets (Hill et al., 2015; Huang et al., 2012).

Predicting Changes: participating systems must predict the change in similarity ratings between the two contexts for each word pair. This will be evaluated using two metrics: binary accuracy of predicting direction of change; and uncentered Pearson correlation, as a measure of accuracy of predicting relative magnitude of changes. We use the uncentered correlation to allow for differences in scaling while maintaining the effect of direction of change. On this subtask, any context-independent

[7] https://www.aclweb.org/portal/content/semeval-2020-international-workshop-semantic-evaluation
[8] http://embeddia.eu/2019/07/17/shared-task-at-semeval-2020-organized-by-embeddia/
[9] See https://competitions.codalab.org/competitions/20905

Word1: population **Word2: people**	**SimLex: μ 7.68 σ 0.80**
Context1	**Context1: μ 6.49 σ 1.40**
Disease also kills off a lot of the gazelle **population**. There are many **people** and domesticated animals that come onto their land. If they pick up a disease from one of these domesticated species they may not be able to fight it off and die. Also, a big reason for the decline of this gazelle population is habitat destruction.	
Context2	**Context2: μ 7.73 σ 1.77**
But the discontent of the underprivileged, landless and the unemployed sections remained even after the reforms. The crumbling industries give rise to extreme unemployment, in addition to the rapidly growing **population**. These **people** mostly belong to the SC/ST or the OBC. In most cases, they join the extremist organizations, mentioned earlier, as an alternative to earn their livelihoods.	

Figure 6: Example word pair with two contexts, also showing mean and standard deviation of human similarity judgements from our pilot study, together with the SimLex equivalent values for comparison. Note that the human perception of similarity changes between the two contexts (it is higher for context 2 than for context 1), even though the target word pair remains the same.

model will predict no change between contexts, and therefore score the same as a random baseline.

Baselines: we will provide five baselines based on cosine distances between word embeddings: standard word2vec embeddings as a context-independent model; context-dependent ELMo and BERT models on their own; and the concatenation of word2vec and ELMo/BERT embeddings.

5.2 Dataset creation

Our primary current focus is the creation of a suitable set of annotated datasets to support this task, and to provide a new standard resource for intrinsic evaluation of context-dependent embeddings across a range of languages. The datasets are based on pairs of words from SimLex-999 (Hill et al., 2015), a standard dataset for evaluation of static, context-independent embeddings; the reliability and common use of this dataset makes it a good starting point and allows comparison of judgements and model outputs to the context-independent case. We are in the process of creating five datasets, one per language. The English dataset will be the largest, consisting of 333 word pairs, with each pair rated within two different contexts; the other languages will take the same format but will be smaller, with 111 word pairs each. To maximize the comparability between the language datasets, the word pairs will be translated into each language, substituting where necessary if sufficient data for context cannot be found. We use existing SimLex translations for Croatian (Mrkšić et al., 2017), Finnish (Venekoski & Vankka, 2017) and Estonian (Kittask, 2019); we have created a Slovenian version following the procedure of Mrkšić et al. (2017) as used for Croatian. The new Slovenian SimLex dataset will be made publicly available via CLARIN (see Section 6).

We use a two step process: in the first step, candidate context paragraphs are chosen to go with each word pair, using a semi-automated procedure; in the second step, human judgements of pairwise word similarity in context are gathered.

For the first step, we have developed a procedure for semi-automatically selecting candidate contexts for later annotation. For each word pair, we find all passages of length 3 sentences in that language's Wikipedia that include both words in the target pair. Then we use both multilingual BERT (Devlin et al., 2019) and our own ELMo models for the target languages (see Section 3.1 above), to extract the (context-specific) embeddings for the target pair of words. Using the cosine distance between those embeddings, we then select the 8 candidates (4 sourced by BERT and 4 sourced by ELMo) with the highest and lowest similarity, with the expectation that these will be likely to produce high and low human ratings of context-specific similarity. We do not expect the human ratings to correspond to the models' similarity scores particularly well in terms of relative or absolute magnitude — in fact, in many cases they disagree strongly — but find this to be a good procedure for selecting contexts in which the perception

of similarity differs. We then pass these examples to a single expert annotator (one for each language, required to be a native speaker), who uses a custom-built interface to select the candidates in which the word pair appear most similar and most different in meaning. These are then used as the contexts for the next step.

In the second step, we gather multiple human judgements of pairwise word similarity, using the same scale as SimLex, and adapting the SimLex annotator instructions in order to benefit from its tested method of explaining how to focus on similarity rather than relatedness or association. Since our interest is in graded change in meaning between the two different contexts, we prioritise a high number of annotators per pair over a high number of pairs. This will help ensure high quality annotations and reduce the effects of noise. For English we crowdsource these annotations, using up to 30 annotators so that problematic cases can be excluded; for the other languages, we recruit annotators directly, and use 12 annotators for each language (and this will be a backup strategy for English if better inter-annotator agreement is needed, although results so far suggest crowdsourcing is suitable).

For each word pair and context, this annotation procedure itself requires two steps. First the annotators will be presented with the paragraph of text, including both words in the target pair (although these will not initially be indicated to the reader). The annotators will be asked to read this paragraph and come up with two words inspired by it. These words can describe the topic, be related to the context or simply come to mind while reading it; the intention of this step is to ensure that the annotators have properly read and considered the paragraph (the results can also be used in data filtering – see below). The reason that the target words are not marked when reading the context paragraph is to help ensure that the the annotators read the complete paragraph, rather than focusing only on the target word pair. Having done that, in the second step the target pair of words are now marked boldface in the context paragraph, and the annotators must rate the similarity between them.

Reliability of annotations will be ensured by an adapted version of SimLex's post-processing, which includes rating calibration, checkpoint questions and the filtering of annotators with very low correlation to the average rating. In addition, we will use responses to the first annotation question to check annotator engagement with the context text and thus filter low quality raters.

So far we have produced trial subsets for English, Slovenian and Croatian, each of 12 word pairs. Inter-annotator agreement is good, with Spearman's correlation ρ 0.78-0.80 for English, 0.78 for Slovenian, and 0.65 for Croatian. The datasets are currently available only to registered SemEval task participants, but will be released publicly once the competition is complete (taking all data from Wikipedia ensures that the dataset can be freely distributed) - see Section 6.

6 Associated outputs

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

Description	URL	Availability
ELMo embeddings	`hdl.handle.net/11356/1277`	Public (GPL v3)
Contextual word analogies	`github.com/EMBEDDIA/contextual-word-analogies`	To become public*
Crosslingual NER	`github.com/EMBEDDIA/crosslingual-NER`	To become public*
Word analogy dataset	`hdl.handle.net/11356/1261`	Public (CC-BY-SA)
SemEval GWSC dataset	`competitions.codalab.org/competitions/20905`	Temporarily restricted†
Slovenian SimLex	`CLARIN TBD`	To become public*

* Resources marked here as "To become public" are available only within the consortium while under development and/or associated with work yet to be published. They will be released publicly when the associated work is completed and published.

† Access to the SemEval dataset is currently restricted to participants in the SemEval task and competition. It will be made public when the competition period is over.

Parts of this work are also described in detail in the following publications, which are attached to this deliverable as appendices:

Citation	Status	Appendix
Ulčar, M., Robnik-Šikonja, M. (2019b). High quality ELMo embeddings for seven less-resourced languages. arXiv preprint arXiv:1911.10049.	Submitted (LREC 2020), available online	Appendix A
Santos Armendariz, C., Purver, M., Ulčar, M., Robnik-Šikonja, M., Pollak, S., Ljubešić, N., Vaik, K. (2019). CoSimLex: A resource for evaluating graded word similarity in context. arXiv preprint arXiv:1912.05320.	Submitted (LREC 2020), available online	Appendix B

7 Conclusions and further work

We describe the the work on contextual and dynamic embeddings performed in T1.2 of the EMBEDDIA project in the first 12 months of project duration. We first explain the changing context of the state of the art: new developments since the EMBEDDIA project was proposed have produced effective models which are both context-dependent and dynamic. We describe some initial work in dynamic embeddings which uses the earlier, static embedding techniques, and see that it has limitations that could be overcome by using the newer techniques. We then present currently the most popular of these newer contextual embeddings, ELMo and BERT. We show that the size of training sets used significantly affects the quality of embeddings produced, and therefore that the existing publicly available ELMo embeddings for EMBEDDIA languages are inadequate. We trained new ELMo embeddings on larger training sets and analysed their properties on the analogy task and the NER task. The results show that our new contextual embeddings produce substantially better results compared to the non-contextual FastText baseline.

As objective intrinsic evaluation of contextual embeddings remains an unsolved issue, we are preparing a novel evaluation dataset, CoSimLex, designed to support a task called Graded Word Similarity in Context (GWSC), currently running as a SemEval 2020 challenge. We describe the objectives of the dataset and its current state.

In our next steps, we will build BERT monolingual embeddings and BERT multilingual embeddings for sensible combinations of EMBEDDIA languages. We will complete the CoSimLex dataset and use it in the evaluation of the newly produced embeddings. Further, we plan to use the contextual embeddings to address practical problems of the news media industry, both by providing them for use within tools developed in work packages WP3-6, and by applying them directly within this work package. Specifically, we will apply dynamic word embeddings to model and discover semantic changes across dimensions of time, genre, domain, topic, and language, giving insight into journalism and culture across the targeted countries.

In future work we will also address dynamic embeddings. In particular, we already started working on leveraging contextual embeddings for detecting diachronic semantic shift.

References

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). Association for Computational Linguistics.

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, June). 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). Minneapolis, Minnesota: Association for Computational Linguistics. Retrieved from `https://www.aclweb.org/anthology/N19-1423` doi: 10.18653/v1/N19-1423

Fares, M., Kutuzov, A., Oepen, S., & Velldal, E. (2017). Word vectors, reuse, and replicability: Towards a community repository of large-text resources. In *Proceedings of the 21st Nordic Conference on Computational Linguistics* (pp. 271–276). Gothenburg, Sweden: Association for Computational Linguistics.

Firth, J. R. (1957). *Papers in linguistics, 1934-1951*. Oxford University Press.

Hamilton, W. L., Leskovec, J., & Jurafsky, D. (2016). Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of acl* (pp. 1489–1501). Retrieved from `http://www.aclweb.org/anthology/P16-1141`

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

Huang, E. H., Socher, R., Manning, C. D., & Ng, A. Y. (2012). Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th annual meeting of the Association for Computational Linguistics: Long papers-volume 1* (pp. 873–882).

Kingma, D. P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd international conference on learning representations (iclr).*

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

Mikolov, T., Le, Q. V., & Sutskever, I. (2013). Exploiting similarities among languages for machine translation. *arXiv preprint 1309.4168*.

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.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing EMNLP* (pp. 1532–1543).

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018, June). 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). New Orleans, Louisiana: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/N18-1202 doi: 10.18653/v1/N18-1202

Pilehvar, M. T., & Camacho-Collados, J. (2018). WiC: 10,000 example pairs for evaluating context-sensitive representations. *arXiv preprint arXiv:1808.09121*.

Purver, M., Valentinčič, A., Pahor, M., & Pollak, S. (2018, May). Diachronic lexical changes in company reports: An initial investigation. In M. El-Haj, P. Rayson, & A. Moore (Eds.), *Proceedings of the 1st financial narrative processing workshop (FNP 2018)* (pp. 23–30). Retrieved from http://lrec-conf.org/workshops/lrec2018/W27/pdf/book_of_proceedings.pdf

Reimers, N., & Gurevych, I. (2019, April). Alternative weighting schemes for elmo embeddings. *arXiv preprint arXiv:1904.02954*. Retrieved from http://tubiblio.ulb.tu-darmstadt.de/112384/

Riedl, E. J. (2004, July). An examination of long-lived asset impairments. *The Accounting Review*, *79*(3), 823–852.

Santos Armendariz, C., Purver, M., Ulčar, M., Robnik-Šikonja, M., Pollak, S., Ljubešić, N., ... Vaik, K. (2019). CoSimLex: A resource for evaluating graded word similarity in context. *arXiv preprint arXiv:1912.05320*. Retrieved from https://arxiv.org/abs/1912.05320

Ulčar, M., & Robnik-Šikonja, M. (2019). High quality ELMo embeddings for seven less-resourced languages. *arXiv preprint arXiv:1911.10049*.

Van der Maaten, L., & Hinton, G. (2008). Visualizing high-dimensional data using t-SNE. *Journal of Machine Learning Research*, *9*(2579-2605), 85.

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, 22-24 may 2017, gothenburg, sweden* (p. 231-236). Linköping University Electronic Press, Linköpings universitet.

High Quality ELMo Embeddings for Seven Less-Resourced Languages

Matej Ulčar, Marko Robnik-Šikonja
University of Ljubljana, Faculty of Computer and Information Science
Večna pot 113, SI-1000 Ljubljana, Slovenia
{matej.ulcar, marko.robnik}@fri.uni-lj.si

Abstract

Recent results show that deep neural networks using contextual embeddings significantly outperform non-contextual embeddings on a majority of text classification task. We offer precomputed embeddings from popular contextual ELMo model for seven languages: Croatian, Estonian, Finnish, Latvian, Lithuanian, Slovenian, and Swedish. We demonstrate that the quality of embeddings strongly depends on the size of training set and show that existing publicly available ELMo embeddings for listed languages shall be improved. We train new ELMo embeddings on much larger training sets and show their advantage over baseline non-contextual FastText embeddings. In evaluation, we use two benchmarks, the analogy task and the NER task.

Keywords: word embeddings, contextual embeddings, ELMo, less-resourced languages, analogy task, named entity recognition

1. Introduction

Word embeddings are representations of words in numerical form, as vectors of typically several hundred dimensions. The vectors are used as an input to machine learning models; for complex language processing tasks these are typically deep neural networks. The embedding vectors are obtained from specialized learning tasks, based on neural networks, e.g., word2vec (Mikolov et al., 2013b), GloVe (Pennington et al., 2014), FastText (Bojanowski et al., 2017), ELMo (Peters et al., 2018), and BERT (Devlin et al., 2018). For training, the embeddings algorithms use large monolingual corpora that encode important information about word meaning as distances between vectors. In order to enable downstream machine learning on text understanding tasks, the embeddings shall preserve semantic relations between words, and this is true even across languages.

Probably the best known word embeddings are produced by the word2vec method (Mikolov et al., 2013c). The problem with word2vec embeddings is their failure to express polysemous words. During training of an embedding, all senses of a given word (e.g., *paper* as a material, as a newspaper, as a scientific work, and as an exam) contribute relevant information in proportion to their frequency in the training corpus. This causes the final vector to be placed somewhere in the weighted middle of all words' meanings. Consequently, rare meanings of words are poorly expressed with word2vec and the resulting vectors do not offer good semantic representations. For example, none of the 50 closest vectors of the word *paper* is related to science[1].

The idea of **contextual embeddings** is to generate a different vector for each context a word appears in and the context is typically defined sentence-wise. To a large extent, this solves the problems with word polysemy, i.e. the context of a sentence is typically enough to disambiguate different meanings of a word for humans and so it is for the learning algorithms. In this work, we describe high-quality models for contextual embeddings, called ELMo (Peters et al., 2018), precomputed for seven morphologically rich, less-resourced languages: Slovenian, Croatian, Finnish, Estonian, Latvian, Lithuanian, and Swedish. ELMo is one of the most successful approaches to contextual word embeddings. At time of its creation, ELMo has been shown to outperform previous word embeddings (Peters et al., 2018) like word2vec and GloVe on many NLP tasks, e.g., question answering, named entity extraction, sentiment analysis, textual entailment, semantic role labeling, and coreference resolution.

This report is split into further five sections. In section 2., we describe the contextual embeddings ELMo. In Section 3., we describe the datasets used and in Section 4. we describe preprocessing and training of the embeddings. We describe the methodology for evaluation of created vectors and results in Section 5.. We present conclusion in Section 6. where we also outline plans for further work.

2. ELMo

Typical word embeddings models or representations, such as word2vec (Mikolov et al., 2013b), GloVe (Pennington et al., 2014), or FastText (Bojanowski et al., 2017), are fast to train and have been pre-trained for a number of different languages. They do not capture the context, though, so each word is always given the same vector, regardless of its context or meaning. This is especially problematic for polysemous words. ELMo (Embeddings from Language Models) embedding (Peters et al., 2018) is one of the state-of-the-art pretrained transfer learning models, that remedies the problem and introduces a contextual component.

ELMo model's architecture consists of three neural network layers. The output of the model after each layer gives one set of embeddings, altogether three sets. The first layer is a CNN layer, which operates on a character level. It is context independent, so each word always gets the same embedding, regardless of its context. It is followed by two biLM layers. A biLM layer consists of two concatenated LSTMs. In the first LSTM, we try to predict the following

[1] This can be checked with a demo showing words corresponding to near vectors computed with word2vec from Google News corpus, available at `http://bionlp-www.utu.fi/wv_demo/`.

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

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.

Although ELMo is trained on character level and is able to handle out-of-vocabulary words, a vocabulary file containing most common tokens is used for efficiency during training and embedding generation. The original ELMo model was trained on a one billion word large English corpus, with a given vocabulary file of about 800,000 words. Later, ELMo models for other languages were trained as well, but limited to larger languages with many resources, like German and Japanese.

2.1. ELMoForManyLangs

Recently, ELMoForManyLangs (Che et al., 2018) project released pre-trained ELMo models for a number of different languages (Fares et al., 2017). These models, however, were trained on a significantly smaller datasets. They used 20-million-words data randomly sampled from the raw text released by the CoNLL 2017 Shared Task - Automatically Annotated Raw Texts and Word Embeddings (Ginter et al., 2017), which is a combination of Wikipedia dump and common crawl. The quality of these models is questionable. For example, we compared the Latvian model by ELMoForManyLangs with a model we trained on a complete (wikidump + common crawl) Latvian corpus, which has about 280 million tokens. The difference of each model on the word analogy task is shown in Figure 1 in Section 5.. As the results of the ELMoForManyLangs embeddings are significantly worse than using the full corpus, we can conclude that these embeddings are not of sufficient quality. For that reason, we computed ELMo embeddings for seven languages on much larger corpora. As this effort requires access to large amount of textual data and considerable computational resources, we made the precomputed models publicly available by depositing them to Clarin repository.

3. Training Data

We trained ELMo models for seven languages: Slovenian, Croatian, Finnish, Estonian, Latvian, Lithuanian and Swedish. To obtain high-quality embeddings, we used large monolingual corpora from various sources for each language. Some corpora are available online under permissive licences, others are available only for research purposes or have limited availability. The corpora used in training datasets are a mix of news articles and general web crawl, which we preprocessed and deduplicated. Below we shortly describe the used corpora in alphabetical order of the involved languages. Their names and sizes are summarized in Table 1.

Croatian dataset include hrWaC 2.1 corpus[2] (Ljubešić and Klubička, 2014), Riznica[3] (Ćavar and Brozović Rončević, 2012), and articles of Croatian branch of Styria media house, made available to us through partnership in a joint project[4]. hrWaC was built by crawling the .hr internet domain in 2011 and 2014. Riznica is composed of Croatian fiction and non-fiction prose, poetry, drama, textbooks, manuals, etc. The Styria dataset consists of 570,219 news articles published on the Croatian 24sata news portal and niche portals related to 24sata.

Estonian dataset contains texts from two sources, CoNLL 2017 Shared Task - Automatically Annotated Raw Texts and Word Embeddings[5] (Ginter et al., 2017), and news articles made available to us by Ekspress Meedia due to partnership in the project. Ekspress Meedia dataset is composed of Estonian news articles between years 2009 and 2019. The CoNLL 2017 corpus is composed of Estonian Wikipedia and webcrawl.

Finnish dataset contains articles by Finnish news agency STT[6], Finnish part of the CoNLL 2017 dataset, and Ylilauta downloadable version[7] (Ylilauta, 2011). STT news articles were published between years 1992 and 2018. Ylilauta is a Finnish online discussion board; the corpus contains parts of the discussions from 2012 to 2014.

Latvian dataset consists only of the Latvian portion of the ConLL 2017 corpus.

Lithuanian dataset is composed of Lithuanian Wikipedia articles from 2018, DGT-UD corpus[8], and LtTenTen[9]. DGT-UD is a parallel corpus of 23 official languages of the EU, composed of JRC DGT translation memory of European law, automatically annotated with UD-Pipe 1.2. LtTenTen is Lithuanian web corpus made up of texts collected from the internet in April 2014 (Jakubíček et al., 2013).

Slovene dataset is formed from the Gigafida 2.0 corpus (Krek et al., 2019). It is a general language corpus composed of various sources, mostly newspapers, internet pages, and magazines, but also fiction and non-fiction prose, textbooks, etc.

Swedish dataset is composed of STT Swedish articles and Swedish part of CoNLL 2017. The Finnish news agency STT publishes some of its articles in Swedish language. They were made available to us through partnership in a joint project. The corpus contains those articles from 1992 to 2017.

4. Preprocessing and Training

Prior to training the ELMo models, we sentence and word tokenized all the datasets. The text was formatted in such a way that each sentence was in its own line with tokens separated by white spaces. CoNLL 2017, DGT-UD and LtTenTen14 corpora were already pre-tokenized. We tokenized the others using the NLTK library[10] and its tokeniz-

[2] http://hdl.handle.net/11356/1064

[3] http://hdl.handle.net/11356/1180

[4] http://embeddia.eu

[5] http://hdl.handle.net/11234/1-1989

[6] http://urn.fi/urn:nbn:fi:lb-2019041501

[7] http://urn.fi/urn:nbn:fi:lb-2016101210

[8] http://hdl.handle.net/11356/1197

[9] https://www.sketchengine.eu/lttenten-lithuanian-corpus/

[10] https://www.nltk.org/

Table 1: The training corpora used. We report their size (in billions of tokens), and ELMo vocabulary size (in millions of tokens).

Language	Corpora	Size	Vocabulary size
Croatian	hrWaC 2.1, Riznica, Styria articles	1.95	1.4
Estonian	CoNLL 2017, Ekspress Meedia articles	0.68	1.2
Finnish	STT articles, CoNLL 2017, Ylilauta downloadable version	0.92	1.3
Latvian	CoNLL 2017	0.27	0.6
Lithuanian	Wikipedia 2018, DGT-UD, LtTenTen14	1.30	1.1
Slovene	Gigafida 2.0	1.26	1.4
Swedish	CoNLL 2017, STT articles	1.68	1.2

ers for each of the languages. There is no tokenizer for Croatian in NLTK library, so we used Slovene tokenizer instead. After tokenization, we deduplicated the datasets for each language separately, using the Onion (ONe Instance ONly) tool[11] for text deduplication. We applied the tool on paragraph level for corpora that did not have sentences shuffled and on sentence level for the rest. We considered 9-grams with duplicate content threshold of 0.9.

For each language we prepared a vocabulary file, containing roughly one million most common tokens, i.e. tokens that appear at least n times in the corpus, where n is between 15 and 25, depending on the dataset size. We included the punctuation marks among the tokens. We trained each ELMo model using default values used to train the original English ELMo (large) model.

5. Evaluation

We evaluated the produced ELMo models for all languages using two evaluation tasks: a word analogy task and named entity recognition (NER) task. Below, we first shortly describe each task, followed by the evaluation results.

5.1. Word Analogy Task

The word analogy task was popularized by Mikolov et al. (2013c). 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: 5 semantic and 10 syntactic. To illustrate a semantic relationship, 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, the two words in a pair have a common stem (in some cases even same lemma), with all the pairs in a given category having the same morphological relationship. 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. That task is then to find the term y corresponding to the relationship "dark : y", with the expected answer being y = darker, that is a comparative form of the adjective dark.

In the vector space, the analogy task is transformed into vector arithmetic and 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 the analogy dataset the analogies are already pre-specified, so we are measuring how close are the given pairs. In the evaluation below, we use analogy datasets for all tested languages based on the English dataset by (Mikolov et al., 2013a) [12]. Due to English-centered bias of this dataset, we used a modified dataset which was first written in Slovene language and then translated into other languages (**?**).

As each instance of analogy contains only four words, without any context, the contextual models (such as ELMo) do not have enough context to generate sensible embeddings. We therefore used some additional text to form simple sentences using the four analogy words, while taking care that their noun case stays the same. For example, for the words "Rome", "Italy", "Paris" and "France" (forming the analogy Rome is to Italy as Paris is to x, where the correct answer is x =France), we formed the sentence "If the word Rome corresponds to the word Italy, then the word Paris corresponds to the word France". We generated embeddings for those four words in the constructed sentence, substituted the last word with each word in our vocabulary and generated the embeddings again. As typical for non-contextual analogy task, we measure the cosine distance (d) between the last word (w_4) and the combination of the first three words ($w_2 - w_1 + w_3$). We use the CSLS metric (Conneau et al., 2018) to find the closest candidate word (w_4). If we find the correct word among the five closest words, we consider that entry as successfully identified. The proportion of correctly identified words forms a statistic called accuracy@5, which we report as the result.

We first compare existing Latvian ELMo embeddings from ELMoForManyLangs project with our Latvian embeddings, followed by the detailed analysis of our ELMo embeddings. We trained Latvian ELMo using only CoNLL 2017 corpora. Since this is the only language, where we trained the embedding model on exactly the same corpora as ELMoForManyLangs models, we chose it for comparison between our ELMo model with ELMoForManyLangs. In other languages, additional or other corpora were used, so a direct comparison would also reflect the quality of the corpora used for training. In Latvian, however, only the size of the training dataset is different. ELMoForManyLangs uses only 20 million tokens and we use the whole corpus of 270 million tokens.

[11] http://corpus.tools/wiki/Onion

[12] http://download.tensorflow.org/data/questions-words.txt

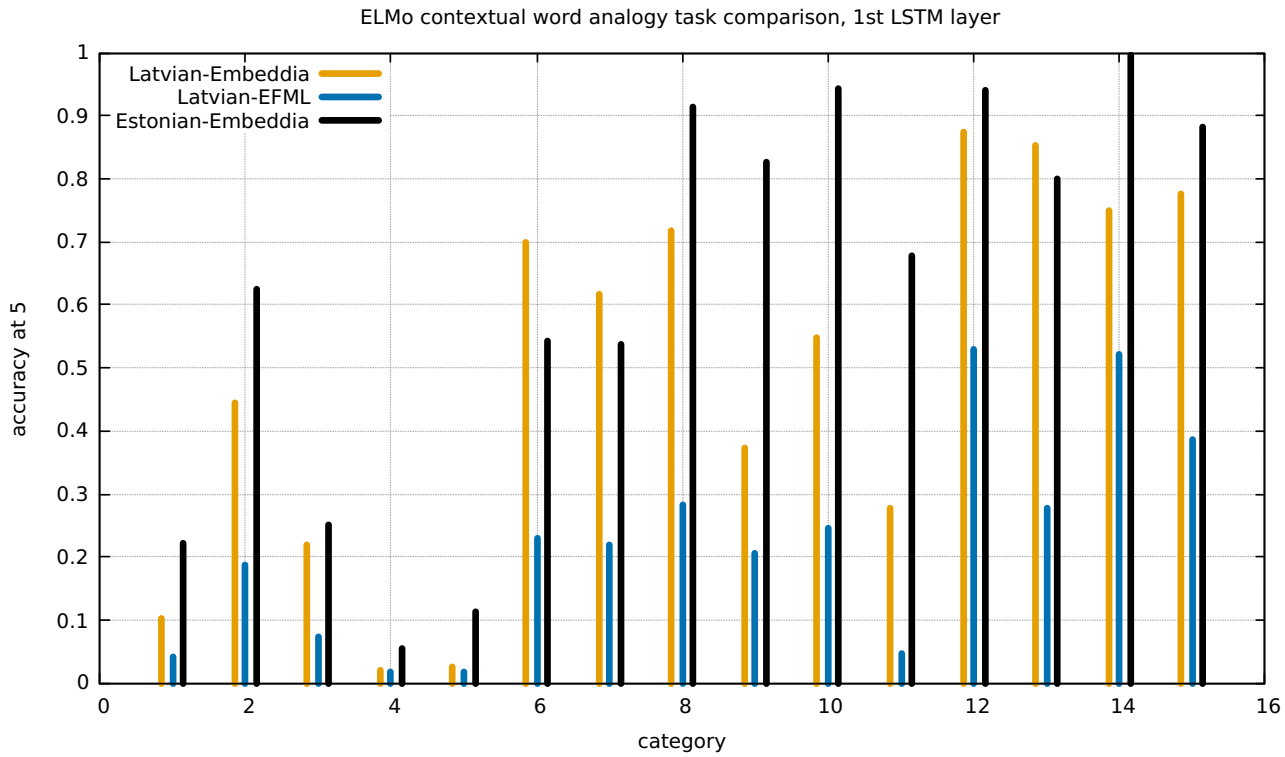

Figure 1: Comparison of Latvian ELMo model by ELMoForManyLangs (blue, latvian-efml), Latvian ELMo model trained by EMBEDDIA team (yellow, latvian-embeddia), and Estonian ELMo model trained by EMBEDDIA team (black, estonian-embeddia). The performance is measured as accuracy@5 on word analogy task, where categories 1 to 5 are semantic, and categories 6 to 15 are syntactic. The embeddings use weights of the first LSTM layer (ie. the second layer overall).

The Latvian ELMo model from ELMoForManyLangs project performs significantly worse than EMBEDDIA ELMo Latvian model on all categories of word analogy task (Figure 1). We also include the comparison with our Estonian ELMo embeddings in the same figure. This comparison shows that while differences between our Latvian and Estonian embeddings can be significant for certain categories, the accuracy score of ELMoForManyLangs is always worse than either of our models. The comparison of Estonian and Latvian models leads us to believe that a few hundred million tokens is a sufficiently large corpus to train ELMo models (at least for word analogy task), but 20-million token corpora used in ELMoForManyLangs are too small.

The results for all languages and all ELMo layers, averaged over semantic and syntactic categories, are shown in Table 2. The embeddings after the first LSTM layer perform best in semantic categories. In syntactic categories, the non-contextual CNN layer performs the best. Syntactic categories are less context dependent and much more morphology and syntax based, so it is not surprising that the non-contextual layer performs well. The second LSTM layer embeddings perform the worst in syntactic categories, though still outperforming CNN layer embeddings in semantic categories. Latvian ELMo performs worse compared to other languages we trained, especially in semantic categories, presumably due to smaller training data size. Surprisingly, the original English ELMo performs very poorly in syntactic categories and only outperforms Latvian in semantic categories. The low score can be partially explained by English model scoring 0.00 in one syntactic category "opposite adjective", which we have not been able to explain.

Table 2: The embeddings quality measured on the word analogy task, using acc@5 score. Each language is represented with its 2-letter ISO code. Results are shown for each layer separately and are averaged over all semantic (sem) and all syntactic (syn) categories, so that each category has an equal weight (i.e. results are first averaged for each category, and these averages are then averaged with equal weights).

Layer	CNN		LSTM1		LSTM2	
Category	sem	syn	sem	syn	sem	syn
hr	0.13	0.79	0.24	0.75	0.20	0.54
et	0.10	0.85	0.25	0.81	0.18	0.63
fi	0.13	0.83	0.33	0.74	0.25	0.54
lv	0.08	0.74	0.16	0.65	0.13	0.43
lt	0.08	0.86	0.29	0.86	0.21	0.62
sl	0.14	0.79	0.41	0.79	0.33	0.57
sv	0.21	0.80	0.25	0.60	0.22	0.34
en	0.18	0.22	0.21	0.22	0.21	0.21

5.2. Named Entity Recognition

For evaluation of ELMo models on a relevant downstream task, we used named entity recognition (NER) task. NER is an information extraction task that seeks to locate and classify named entity mentions in unstructured text into pre-defined categories such as the person names, organizations, locations, medical codes, time expressions, quantities, monetary values, percentages, etc. To allow comparison of results between languages, we used an adapted version of this task, which uses a reduced set of labels, available in NER datasets for all processed languages. The labels in the used NER datasets are simplified to a common label set of three labels (person - PER, location - LOC, organization - ORG). Each word in the NER dataset is labeled with one of the three mentioned labels or a label 'O' (other, i.e. not a named entity) if it does not fit any of the other three labels. The number of words having each label is shown in Table 3.

Table 3: The number of words labeled with each label (PER, LOC, ORG) and the density of these labels (their sum divided by the number of all words) for datasets in all languages.

Language	PER	LOC	ORG	density
Croatian	3931	656	1138	0.064
Estonian	8490	6326	6149	0.096
Finnish	3402	2173	11258	0.087
Latvian	5615	2643	3341	0.085
Lithuanian	2101	2757	2126	0.076
Slovenian	4478	2460	2667	0.049
Swedish	3976	1797	1519	0.047
English	17050	12316	14613	0.146

To measure the performance of ELMo embeddings on the NER task we proceeded as follows. We embedded the text in the datasets sentence by sentence, producing three vectors (one from each ELMo layer) for each token in a sentence. We calculated the average of the three vectors and used it as the input of our recognition model. The input layer was followed by a single LSTM layer with 128 LSTM cells and a dropout layer, randomly dropping 10% of the neurons on both the output and the recurrent branch. The final layer of our model was a time distributed softmax layer with 4 neurons.

We used ADAM optimiser (Kingma and Ba, 2014) with the learning rate 0.01 and 10^{-5} learning rate decay. We used categorical cross-entropy as a loss function and trained the model for 3 epochs. We present the results using the Macro F_1 score, that is the average of F_1-scores for each of the three NE classes (the class Other is excluded).

Since the differences between the tested languages depend more on the properties of the NER datasets than on the quality of embeddings, we can not directly compare ELMo models. For this reason, we take the non-contextual fastText embeddings[13] as a baseline and predict named entities using them. The architecture of the model using fastText embeddings is the same as the one using ELMo embeddings, except that the input uses 300 dimensional fastText embedding vectors, and the model was trained for 5 epochs (instead of 3 as for ELMo). In both cases (ELMo and fastText) we trained and evaluated the model five times, because there is some random component involved in initialization of the neural network model. By training and evaluating multiple times, we minimise this random component.

The results are presented in Table 4. We included the evaluation of the original ELMo English model in the same table. NER models have little difficulty distinguishing between types of named entities, but recognizing whether a word is a named entity or not is more difficult. For languages with the smallest NER datasets, Croatian and Lithuanian, ELMo embeddings show the largest improvement over fastText embeddings. However, we can observe significant improvements with ELMo also on English and Finnish, which are among the largest datasets (English being by far the largest). Only on Slovenian dataset did ELMo perform slightly worse than fastText, on all other EMBEDDIA languages, the ELMo embeddings improve the results.

Table 4: The results of NER evaluation task, averaged over 5 training and evaluation runs. The scores are average F_1 score of the three named entity classes. The columns show FastText, ELMo, and the difference between them ($\Delta(E - FT)$).

Language	FastText	ELMo	$\Delta(E - FT)$
Croatian	0.17	0.53	0.36
Estonian	0.26	0.31	0.05
Finnish	0.71	0.84	0.13
Latvian	0.39	0.45	0.06
Lithuanian	0.43	0.65	0.22
Slovenian	0.68	0.67	-0.01
Swedish	0.82	0.88	0.06
English	0.28	0.43	0.15

6. Conclusion

We prepared precomputed ELMo contextual embeddings for seven languages: Croatian, Estonian, Finnish, Latvian, Lithuanian, Slovenian, and Swedish. We present the necessary background on embeddings and contextual embeddings, the details of training the embedding models, and their evaluation. We show that the size of used training sets importantly affects the quality of produced embeddings, and therefore the existing publicly available ELMo embeddings for the processed languages are inadequate. We trained new ELMo embeddings on larger training sets and analysed their properties on the analogy task and on the NER task. The results show that the newly produced contextual embeddings produce substantially better results compared to the non-contextual fastText baseline. In future work, we plan to use the produced contextual embeddings on the problems of news media industry. The pretrained ELMo models will be deposited to the CLARIN reposi-

[13] https://fasttext.cc/

tory[14] by the time of the final version of this paper.

Acknowledgments

The work was partially supported by the Slovenian Research Agency (ARRS) core research programme P6-0411. This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media). The results of this publication reflects only the authors' view and the EU Commission is not responsible for any use that may be made of the information it contains.

7. Bibliographical References

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Ćavar, D. and Brozović Rončević, D. (2012). Riznica: The Croatian Language Corpus. *Prace filologiczne*, 63:51–65.

Che, W., Liu, Y., Wang, Y., Zheng, B., and 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*, pages 55–64, Brussels, Belgium, October. Association for Computational Linguistics.

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

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Fares, M., Kutuzov, A., Oepen, S., and Velldal, E. (2017). Word vectors, reuse, and replicability: Towards a community repository of large-text resources. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 271–276, Gothenburg, Sweden, May. Association for Computational Linguistics.

Ginter, F., Hajič, J., Luotolahti, J., Straka, M., and Zeman, D. (2017). CoNLL 2017 shared task - automatically annotated raw texts and word embeddings. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Jakubíček, M., Kilgarriff, A., Kovář, V., Rychlý, P., and Suchomel, V. (2013). The tenten corpus family. *7th International Corpus Linguistics Conference CL 2013*, 07.

Kingma, D. P. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

Krek, S., Erjavec, T., Repar, A., Čibej, J., Arhar, S., Gantar, P., Kosem, I., Robnik, M., Ljubešić, N., Dobrovoljc, K., Laskowski, C., Grčar, M., Holozan, P., Šuster, S., Gorjanc, V., Stabej, M., and Logar, N. (2019). Gigafida 2.0: Korpus pisne standardne slovenščine. viri.cjvt.si/gigafida.

Ljubešić, N. and Klubička, F. (2014). bs,hr,srWaC - web corpora of Bosnian, Croatian and Serbian. In *Proceedings of the 9th Web as Corpus Workshop (WaC-9)*, pages 29–35, Gothenburg, Sweden, April. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint 1301.3781*.

Mikolov, T., Le, Q. V., and Sutskever, I. (2013b). Exploiting similarities among languages for machine translation. *arXiv preprint 1309.4168*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013c). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Ylilauta. (2011). The Downloadable Version of the Ylilauta Corpus. http://urn.fi/urn:nbn:fi:lb-2016101210.

[14] TBD

CoSimLex: A Resource for Evaluating Graded Word Similarity in Context

Carlos Santos Armendariz[*], **Matthew Purver**[*†],
Matej Ulčar[‡], **Senja Pollak**[†], **Nikola Ljubešić**[†], **Mark Granroth-Wilding**[◇]
[*]Cognitive Science Research Group, Queen Mary University of London, London, UK
{c.santosarmendariz, m.purver}@qmul.ac.uk
[†]Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia
{senja.pollak, nikola.ljubesic}@ijs.si
[‡]Faculty of Computer and Information Science, University of Ljubljana, Slovenia
matej.ulcar@fri.uni-lj.si
[◇]Department of Computer Science, University of Helsinki, Finland
mark.granroth-wilding@helsinki.fi

Abstract

State of the art natural language processing tools are built on context-dependent word embeddings, but no direct method for evaluating these representations currently exists. Standard tasks and datasets for intrinsic evaluation of embeddings are based on judgements of similarity, but ignore context; standard tasks for word sense disambiguation take account of context but do not provide continuous measures of meaning similarity. This paper describes an effort to build a new dataset, CoSimLex, intended to fill this gap. Building on the standard pairwise similarity task of SimLex-999, it provides context-dependent similarity measures; covers not only discrete differences in word sense but more subtle, graded changes in meaning; and covers not only a well-resourced language (English) but a number of less-resourced languages. We define the task and evaluation metrics, outline the dataset collection methodology, and describe the status of the dataset so far.

Keywords: corpus, annotation, semantics, similarity, context, salience, context-dependence

1. Introduction

Recent work in language modelling and word embeddings has led to a sharp increase in use of context-dependent models such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019). These models, by providing representations of words which depend on the surrounding context, allow us to take account of the effects not only of discrete differences in word sense but of the more graded effects of context. However, evaluation of these models has generally been in terms of either their performance as language models, or their effect on downstream tasks such as sentiment classification (Peters et al., 2018): there are few resources available which allow evaluation in terms of the properties of the embeddings themselves, or in terms of their ability to model human perceptions of meaning. There are established methods to evaluate word embedding models intrinsically via their ability to reflect human similarity judgements (see e.g. WordSim-353 (Finkelstein et al., 2002) and SimLex-999 (Hill et al., 2015)) or model analogies (Mikolov et al., 2013); however, these have generally ignored context and treated words in isolation. The few that do provide context (e.g. SCWS (Huang et al., 2012) and WiC (Pilehvar and Camacho-Collados, 2019)) focus on word sense and discrete effects, thus missing some of the effects that context has on words in general, and some of the benefits of context-dependent models. To evaluate current models, we need a way to evaluate their ability to reflect similarity judgements *in context*: how well do they model the sometimes subtle effects that context has on word meaning?

In this paper we present our ongoing efforts to define and build a new dataset that tries to fill that gap: **CoSimLex**. CoSimLex builds on the familiar pairwise, graded similarity task of SimLex-999, but extends it to pairs of words as they occur in context, and specifically provides two different shared contexts for each pair of words. This will provide a dataset suitable for intrinsic evaluation of state-of-the-art contextual word embedding models, by testing their ability to reflect human judgements of word meaning similarity in context, and crucially, the way in which this varies as context is changed. It goes beyond other existing context-based datasets by taking the *gradedness* of human judgements into account, thus applying not only to polysemous words, or words with distinct senses, but to the phenomenon of context-dependency of word meaning in general. In addition the new dataset is multi-lingual, and includes four less-resourced European languages: Croatian, Estonian, Finnish and Slovene.

The dataset will be used as the gold standard for the final evaluation of a currently running task at SemEval2020: **Task 3 Graded Word Similarity in Context**.[1]

2. Background

From the outset, our main motivation for the development of this dataset came from an interest in the cognitive and psychological mechanisms by which context affects our perception of the meaning of words. There have been many different ways in the literature to look at this phenomenon, which lie in the intersection of several different fields of research, and a detailed discussion of the different approaches to this problem is out of the scope of this paper; here, we present two of the most prominent ideas that helped define what we were trying to capture, and made an impact in the

[1]https://competitions.codalab.org/competitions/20905

design of the dataset and its annotation process. We then look at previous datasets that deal with similarity in context.

2.1. Contextual Modulation

Within the field of lexical semantics, Cruse (1986) proposed an interesting compromise between those linguists that saw words as associated with a number of discrete senses and those that thought that the perceived discreteness of lexical senses is just an illusion. He distinguishes two different manners in which sentential context modifies the meaning of a word. First, the context can select for different discrete senses; if that is the case, the word is described as *ambiguous*, and the process is referred as **contextual selection of senses**. The second way works within the scope of a single sense, modifying it in an unlimited number of ways by *highlighting* certain semantic traits and *backgrounding* others. This process is called **contextual modulation of meaning**, and the word is said to be *general* with respect to the traits that are being modulated. This effect is by nature not discrete but continuous and fluid, and since every word is *general* to some extent: it can be argued that a word has a different meaning in every context in which it appears.
Some examples can help to see the different ways in which these phenomena work in real life:

1. We finally reached the bank.
2. At this point, the bank was covered with brambles.
3. Sue is visiting her pregnant cousin.
4. Arthur poured the butter into a dish.

In the first sentence the context doesn't really help us to select a sense for the word *bank*. This creates some tension: because bank is such an ambiguous word, we need to select a sense in order for the sentence to properly work. This is an example of *ambiguity* as opposed to *generality*. In the second sentence one of the senses is clearly more *normal* than the other. Cruse (1986) sees the evaluation of *contextual normality* as the main mechanism for sense selection. In the third sentence, the word *cousin* could in principle refer to a male or a female. The context is clearly telling us that we are talking about a female cousin, however in this case *cousin* is a *general* word that includes male, female, but as well tall, short, happy and sad cousins. The meaning of *cousin* is being *modulated* by the context to promote the "female" trait; but notice that the sentence "Sue is visiting her cousin" doesn't create any tension: *cousin* is not ambiguous in the true sense. The last sentence is another example of *contextual modulation* highlighting the "liquid" trait for *butter*. It is interesting to notice that in this case not only "liquid" is highlighted, related traits like "warm" can be highlighted as a consequence.
These two processes happen very commonly together, with the same context forcing a sense and then modulating its expression. Many different explanations have been proposed for the emergence of these discrete senses, and some may have their origins in very commonly modulated meaning but, according to Cruse, once a discrete sense is established it become some different to *contextual modulation* and follows different rules:

5. John prefers bitches to dogs.
6. ? John prefers bitches to canines.
7. ? Mary likes mares better than horses.

Here the first sentence works because one of the discrete senses associated to the word *dog* refers only to male dogs. This cannot be explained by *contextual modulation*. If that was the case the second sentence, which replaces *dog* with *canine*, would work and *canine* would be modulated in the same way than *dog* was. The fact that neither *canine* nor *horse* can be modulated in this same way indicates that meaning modulation and sense selection are two, strongly interconnected, but distinctive mechanisms of contextual variability.
Given this, it seems clear that the contextual selection of senses would modify human judgements of similarity. For example, the word *bank*, when used in a context which selects its financial institution sense, should be scored as more similar to other kinds of financial institution (e.g. *building society*) than when in a context which selects the geographic sense of the word. However, we should also expect that a word like *butter*, when contextually modulated to highlight its "liquid", "hot" and "frying" traits, should score more similar to *vegetable oil* than when contextually modulated to highlight its "animal sourced", "dairy", and "creamy" traits. This kind of hypothesis would be testable given a new context-dependent similarity dataset.
Interestingly, Cruse doesn't find the contrast between polysemy and homonymy particularly helpful, and dislikes the use of these terms because they promote the idea that the primary semantic unit is some common lexeme and each of the different senses are just variants of it. He instead believes the primary semantic unit should be the *lexical units*, a union of a single sense and a lexical form, and finds it more useful to look at the contrast between discrete and continuous semantic variability. It is true that homonymous words will always fall into the discrete category, but most common understandings of polysemy would include both discrete and continuous variations.

2.2. Salience Manipulation

Until now we have looked at contextual variability as an exclusively linguistic phenomenon, a point of view rooted in lexical semantics. We looked at how the context of the sentence affects the meaning of the word. In contrast, cognitive linguistics, and the more specific cognitive semantics, look at language and meaning as an expression of human cognition more generally (Evans and Green, 2018).
This approach champions concepts, more specifically *conceptual structures*), as the true recipient of meaning, replacing words or lexical units. These linguistic units no longer refer to objects in an external world but to concepts in the mind of the speaker. Words get their meaning only by association with *conceptual structures* in our minds. The process by which we construct meaning is called conceptualisation, an embodied phenomenon based in social interaction and sensory experience.

Cognitive linguists gravitate to themes that focus on the flexibility and the ability of the interaction between language and conceptual structures to model continuous phenomena, like prototyping effects, categories, metaphor theory and new ways to look at polysemy. Within the cognitive tradition, the idea of *conceptual spaces*, characterised by *conceptual dimensions*, has been especially influential (Gärdenfors, 2000; Gärdenfors, 2014). These dimensions can range from concrete ones like weight, temperature and brightness, to very abstract ones like awkwardness or goodness. Once a domain, or selection of dimensions is established, a concept is defined as a region (usually a convex one) of the conceptual space. An example would be to define the colour *brown* as a region of a space made of the dimensions *Red*, *Green* and *Blue*. This geometric approach lends itself perfectly to model phenomena like prototyping (central point of the region), similarity (distance), metaphor (projection between different dimensions) and, more importantly for our concerns here, fluid changes in meaning due to the effects of context.

Warglien and Gärdenfors (2015) use conceptual spaces to look at *meaning negotiation* in conversation. They investigate the mechanisms, consciously or unconsciously, employed by the people involved in conversation to negotiate meaning of vague predicates, in order to satisfy the coordination needed for communication. These tools help them to decide areas in which they don't agree as well. All these processes work by manipulating the conceptual dimensions in which meaning is represented. We will refer to them as **salience manipulation** because their main role is to dynamically rise or lower the perceived importance of certain conceptual dimensions.

The main mechanism by which speakers can modify salience of conceptual dimensions are the automatic *priming* effects described by, for example, Pickering and Garrod (2004): mentioning specific words early in the conversation can make the dimensions associated with such words more relevant. Speakers can also explicitly try to remove dimensions from the domain in order to promote agreement, or bring in new dimensions by using *metaphoric projections*. Because metaphors can be understood as mappings that transfer structure from one domain to another, they can introduce new dimensions and meaning to the conversation.

> The lion Ulysses emphasizes Ulysses' courage but hides his condition of a castaway in Ogiya. Thus metaphors act by orienting communication and selecting dimensions that may be more or less favorable to the speaker. By suggesting that a storm hit the financial markets, a bank manager can move the conversation away from dimensions pertaining to his own responsibilities and instead focus on dimensions over which he has no control. (Warglien and Gärdenfors, 2015)

From this perspective, then, the change in meaning is no longer a change in the meaning of a specific word, but a change in the mind of the hearer (or reader), a change in their *mental state* triggered by their interaction with the context. In addition, the expectation that priming is the main mechanism for modifying salience has its own implications: Branigan et al. (2000) found that priming effects are much stronger in the context of as natural dialog as possible, when speakers had no time constraints and could respond at their own pace.

This has implications for the design of our dataset and annotation methodology: it is crucial for us to create an annotation process in which the annotator interacts with the context, and does so in as natural a way as possible, before they rate the similarity. Because priming is an automatic process, them knowing that they should be annotating similarity in context becomes a lot less important.

One last interesting consequence of looking at this type of contextual effect is that because the change is in the mind of the annotator, the words that we are rating don't need to be part of the context. From the classical lexical semantics perspective, meaning change comes from the interaction between the word and the rest of the context; but the cognitive approach suggests that if the context triggers changes in the salience of conceptual dimensions related to particular words being annotated, we should see change in the scoring of similarity even if those words are not explicitly present in the context. Our goal in this dataset is therefore to create an annotation process that allows us to capture both of these possible contextual phenomena.

2.3. Existing Datasets

There are a few examples of datasets which take context into account. However, so far these have been motivated by discrete *sense disambiguation*, and therefore take a view of word meaning as discrete (taking one of a finite set of senses) rather than continuous; they are therefore not suited for the more graded effects we are interested to look into.

The **Stanford Contextual Word Similarity (SCWS)** dataset (Huang et al., 2012) does contain graded similarity judgements of pairs of words in the context of organically occurring sentences (from Wikipedia). However it was designed to evaluate a discrete multi-prototype model, so the focus was on the contexts selecting for one of the word senses. This resulted in them presenting each of the two words of the pair in their own distinct context. From our point of view this approach has some drawbacks: First, even in the cases where they annotated the same pair twice, we find ourselves with four different contexts, each affecting the meaning of each of the instances of the words independently, and it is not possible to produce a systematic comparison of contextual effects on pairwise similarity. Second, beyond the independent lexical semantics of each word being affected by their independent *local context*, the annotator is being presented with two completely independently occurring contexts at the same time. Even if the two context did organically occur on their own, this combination of the two didn't, and we have seen before how crucial we think keeping the interaction with the context as natural as possible is. There is no easy way to know how this newly assembled *global context* affects the cognitive state of the annotators and their perception of similarity. The same goes for the contextually-aware models trying to predict their results. Joining the contexts before feeding them to the model could create conflicting, difficult to predict ef-

Word1: population **Word2: people**	**SimLex: μ 7.68 σ 0.80**
Context1	**Context1: μ 6.49 σ 1.40**
Disease also kills off a lot of the gazelle **population**. There are many **people** and domesticated animals that come onto their land. If they pick up a disease from one of these domesticated species they may not be able to fight it off and die. Also, a big reason for the decline of this gazelle population is habitat destruction.	
Context2	**Context2: μ 7.73 σ 1.77**
But the discontent of the underprivileged, landless and the unemployed sections remained even after the reforms. The crumbling industries give rise to extreme unemployment, in addition to the rapidly growing **population**. These **people** mostly belong to the SC/ST or the OBC. In most cases, they join the extremist organizations, mentioned earlier, as an alternative to earn their livelihoods.	

Figure 1: Example from the English pilot, showing a word pair with two contexts, mean and standard deviation of human similarity judgements and the original SimLex equivalent values for comparison.

fects, but feeding each context independently is fundamentally different to what humans annotators were presented with.

In addition to these limitations of the independent contexts approach, the scores found in SCWS show a worryingly low inter-rater agreement (IRA), measured as the Spearman correlation between different annotators. As pointed out by (Pilehvar and Camacho-Collados, 2019), the mean IRA between each annotator and the average of the rest, which is considered a human-level upper bound for model's performance, is 0.52; while the performance of a simple context-independent model like word2vec (Mikolov et al., 2013) is 0.65. Examining the scores more in detail, we find that many scores show a very large standard deviation, with annotators rating the same pair very differently. One possible reason for this may lie in the annotation design: the task itself does not directly enforce engagement with the context, and the words were presented to annotators highlighted in boldface, making it easy to pick them out from the context without reading it; thus potentially leading to a lack of engagement of the annotators with the context.

A lot of these limitations were addressed by the more recent **Words-in-Context (WiC)** dataset (Pilehvar and Camacho-Collados, 2019). With a more direct and straightforward take on word sense disambiguation, each entry of the dataset is made of two lexicographer examples of the same word. The entry is completed with a positive value (T) if the word sense in the two examples/context is the same, or with a negative value (F) if the contexts point to different word senses. One advantage of this design is that it forces engagement with the context; another is that it creates a task in which context-independent models like word2vec "would perform no better than a random baseline". Human annotators are shown to produce healthy inter-rater agreement scores for this dataset. However the dataset is again focused in looking at discrete word senses and cannot therefore capture continuous effects of context in the judgements of similarity between different words.

These datasets are also available only in English, and thus do not allow models to be evaluated across different languages.

3. Dataset and Task Design

CoSimLex will be based on pairs of words from SimLex-999 (Hill et al., 2015); the reliability and common use of this dataset makes it a good starting point and allows comparison of judgements and model outputs to the context-independent case. For Croatian, Estonian and Finnish we are using existent translations of Simlex-999 (Mrkšić et al., 2017; Venekoski and Vankka, 2017; Kittask, 2019). In the case of Slovene, we have produced our own new translation, following the methodology used by Mrkšić et al. (2017) for Croatian.

The English dataset consists of 333 pairs; the Croatian, Estonian, Finnish and Slovene datasets of 111 pairs each. Each pair is rated within two different contexts, giving a total of 1554 scores of contextual similarity. This poses a difficult task: to find suitable, organically occurring contexts for each pair; this task is more pronounced for languages with less resources, and as a result the selection of pairs is different for each language.

Each line of CoSimLex will be made of a pair of words selected from Simlex-999; two different contexts extracted from Wikipedia in which these two words appear; two scores of similarity, each one related to one of the contexts; and two scores of standard deviation. Please see Figure 1 for an example from our English pilot.

Evaluation Tasks and Metrics The first practical use of CoSimLex will be as a gold standard for the public SemEval 2020 task 3: *Graded Word Similarity in Context*. The goal of this task is to evaluate how well modern context-dependent embeddings can predict the effect of context in human perception of similarity. In order to do so we define two subtasks and two metrics:

Subtask 1 - Predicting Changes: In subtask 1, participants must predict the *change* in similarity ratings between the two contexts. In order to evaluate it we calculate the difference between the scores produced by the model when the pair is rated within each one of the two contexts. We do the same with the average of the scores produced by the human annotators. Finally we calculate the uncentered Pearson correlation. A key property of this method is that any context-independent model will predict no change and get strongly penalised in this task.

Subtask 2 - Predicting Ratings: In subtask 2, participants must predict the absolute similarity rating for each pair in each context. This will be evaluated using Spearman correlation with gold-standard judgements, following the standard evaluation methodology for similarity datasets (Hill et al., 2015; Huang et al., 2012). Good

context-independent models could theoretically give competitive results in this task, however we still expect context-dependent models to have a considerable advantage.

4. Annotation Methodology

As starting point for our annotation methodology, we adapted the annotation instructions used for SimLex-999. This way we benefit from its tested method of explaining how to focus on *similarity* rather than *relatedness* or *association* (Hill et al., 2015). For English we adopted a modified version of their crowd-sourcing process: we use *Amazon Mechanical Turk*, with the same post-processing and cleaning of the data (a necessary step when working with this kind of crowd-sourcing platform), and achieve similarly good inter-annotator agreement. For the less-resourced languages, crowdsourcing is not a viable option due to lack of available speakers, and we recruit annotators directly. This means fewer annotators (for Croatian and Slovene, 12 annotators vs 27 in English), however the average quality of annotation is a lot higher and the data requires less post-processing - see Section 5. for details.

4.1. Finding Suitable Contexts

For each word pair we need to find two suitable contexts. These contexts are extracted from each language's Wikipedia. They are made of three consecutive sentences and they need to contain the pair of words, appearing only once each. English is by far the easiest language to work with, not only because of the amount and quality of the text contained in the English version of Wikipedia but because the other four languages are highly inflected (Croatian, Estonian, Finnish and Slovene). In order to overcome this we work with data from (Ginter et al., 2017)[2] which contains tokenised and lemmatised versions of Wikipedia for 45 languages.

We first find all the possible candidate contexts for each word pair, and then select those candidates that are most likely to produce different ratings of similarity. The differences are expected to be small, especially in words that don't present several senses and are not highly polysemous, so we need a process that has the most chances of finding contexts that make a difference. We use a dual process in which we use ELMo and BERT to rate the similarity between the target pair within each of the candidate contexts. Then we select the 2 contexts in which ELMo scored the pair as the most similar, and the 2 contexts in which it scored them as most different. We do the same using BERT scores. This gives us 4 contexts in which our target words are scored as very similar by the models and 4 contexts in which they are scored as very different.

The final selection of two contexts is made by expert human annotators, one per language. We construct online surveys with these 8 contexts and ask them to select the two in which they think the word pair is the most and the least similar, trying to maximise the potential contrast in similarity. In addition, we ask them how much potential for a difference they see in the contexts selected. This gives us not only the contexts we need, but a predicted performance and direction of change for use in later analysis.

[2]http://hdl.handle.net/11234/1-1989

In the case of less resourced languages, the smaller size and lower quality of the Wikipedia text resources require some extra steps to ensure the quality of the final annotation. For these languages we run the contexts through a set of heuristic filters to try to remove badly constructed ones. In addition we produce 16 candidates instead of 8 for the expert annotators to choose from, and we add the possibility for them to delete parts of the context in order to make them easier to read. Adding text is not allowed, in order to ensure that contexts are natural.

4.2. Contextual Similarity Annotation

The next step is to obtain the contextualised similarity annotations. Our goal is to capture the kind of contextual phenomena discussed in Section 2.: lexical meaning modulation and conceptual salience manipulation. In order to maximise our chances we define three goals:

- We want the interaction with the context to be as natural as possible, so as to maximise priming effects and capture the potential change in the salience of conceptual dimensions.
- We need a way in which annotators have the chance to account for lexical modulation within the sentence.
- We need to avoid the apparent lack of engagement we saw in the SCWS annotators.

With these goals in mind we designed a two-step mixed annotation process. Our online survey interface is composed of two pages per pair of words and context (each annotator scores only one of the contexts). In the first page the annotators are presented with the context, and asked to read it and come up with two words "inspired by it". Once this is complete, the second page shown presents the context again, but with the pair of words now highlighted in bold; they are now asked to rate the similarity of the pair of words within the sentence.

The second page is the main scoring task; it is designed to capture changes in scores of similarity due both to lexical modulation and — because we hope the annotators are still primed by their recent previous engagement with the context — the changes in the salience of conceptual dimensions. The separate task on the first page is intended to make annotators engage fully with the whole context, while maintaining a natural interaction with it to maximise any priming effects. One of the possible problems we identified in the the SCWS annotation process is the fact that the words were always highlighted in bold, making it easy for annotators (Amazon Mechanical Turk workers) to just look at the pair of words in isolation and to not read the rest of the contexts. Our initial task is designed to prevent this (the words are not bold in the first page).

In English, given the resources available, we follow SimLex-999 closely: we will use Amazon Mechanical Turk to get 27 annotators per pair and context. Annotators do not score the same pair twice: 27 annotators score the pair within one context and another 27 in the other. This means the whole dataset can be annotated at the same time. Reliability of annotations will be ensured by an adapted version of SimLex-999's post-processing, which includes rat-

Word1: čovjek (adult male) **Word2: dijete (child)**	
Context1	**Context1: μ 2.5 σ 1.76**
Špinat ima dosta željeza, ali i oksalne kiseline. Oksalna kiselina veže kalcij i čini ga neupotrebljivim za ljudski organizam. Prema novijim istraživanjima, špinat se ne preporuča kao česta hrana mlađim osobama i **djeci**, ali je izvrsna hrana za starije **ljude**. (Spinach has plenty of iron but also oxalic acid. Oxalic acid binds calcium and renders it unusable for the human body. According to recent research, spinach is not recommended as a common food for young people and **children**, but it is an excellent food for older **people**.)	
Context2	**Context2: μ 4.25 σ 0.95**
Nakon što su **ljudi** u selu saznali da je trudna, počinju sumnjati na dr. Richardsona jer je on proveo najviše vremena s njom. Kako vrijeme prolazi, pritisak glasina na kraju prisiljava liječnika da se preseli. Odluči se oženiti s Belindom i uzeti **dijete** sa sobom. (After **people** in the village find out she is pregnant, they begin to suspect Dr. Richardson because he spent the most time with her. As time goes on, the pressure of the rumors eventually forces the doctor to move. She decides to marry Belinda and take her **child** with her.)	

Figure 2: Example from the Croatian pilot (translated to English using Google Translate), showing the word pair with two contexts, mean and standard deviation of human similarity judgements. This example showed one of the most significant contextual effects in the pilot; it went in the opposite direction to the one predicted by the expert annotator. Note the effect of stemming: the target word *čovjek* appears in both cases via its irregular plural, *ljudi* (nominative) or *ljude* (accusative); and *dijete* appears in Context 1 in its dative plural form *djeci*.

ing calibration and the filtering of annotators with very low correlation to the average rating. In addition, we will use responses to the first annotation question to check annotator engagement with the context text and thus filter low quality raters.

For Croatian, Estonian, Finnish and Slovene we recruit annotators directly: this means we have less of them (12 vs 27) but we expect the quality of the annotation to be better (and pilots confim this – see below). It also means, howeve, that we must use the same annotators to rate the two contexts of each pair. This has an avantage, because it controls for the variation in the particular judgement of different annotators, but means that we introduce a week's delay in between annotations in order to make sure they don't remember, and are influenced by, their own previous score.

5. Current Status

Methodology prototyping We have run three pilots with 13 pairs of words each to confirm the annotation design and methodology. Each study tested a slight variation: in the first pilot, annotators rated *relatedness* in addition to similarity; the second focused on similarity, and tested the use of contexts related to the target words but not containing them; the third experimented with marking the target words in the context paragraphs using boldface font.

The first pilot confirmed that (as with SimLex) similarity is a more useful metric for this task than relatedness, displaying a higher inter-annotator agreement and more variation between contexts; we therefore use similarity as the basis of our dataset, as described above.

The results of the second pilot saw significant contextual effects in many examples, including some in which the target words weren't included in the contexts. This indicates that our method seems suitable for capturing priming effects and salience manipulation, or at least some kind of cognitive effects different from lexical contextualisation.

The third pilot showed much lower agreement and lower difference between contexts: we take this as confirmation of our suspicion (from analysis of SCWS) that marking the target words makes it easy for annotators to ignore the rest of the context paragraph, and therefore use the two-stage annotation methodology described above, in which target words are *not* initially marked.

Results The results from tests so far are very promising in terms of both the difference in judgements between contexts, and inter-annotator agreement. In the English pilot with the closest design to the current one (the second pilot described above), we collected 27 different ratings for each pair and each context: see Table 1 for detailed results. In addition to the English pilots we have run two pilots in Croatian and Slovene. Please see Table 2 and Figure 2 for the general results of the Croatian pilot and one of the best examples that came from it respectively.

Inter-rater agreement (IRA) was measured as Spearman correlation between each rating and the average: for the English, pilot, the mean was $\rho = 0.79$, with average standard deviation $\sigma = 1.6$; these compare well to other related datasets (SimLex-999 $\rho = 0.78$, SCWS $\rho = 0.52$). IRA was very high for the Slovene pilot $\rho = 0.82$; significantly lower but still reasonable for the Croatian one $\rho = 0.68$.

In the English pilot, about a third of the pairs show a significant difference in the ratings between contexts, as assessed by a Mann-Whitney U test at $p < 0.05$. The Slovene and Croatian pilots are very small (6 annotators per pair/context) and it is currently difficult to know how significant their results are (but see Table 2 for indications as to the most likely differences); they have however provided invaluable feedback on methods required for the particularities of these highly inflected, less-resourced languages.

At the moment of writing this paper we are preparing to run a second round of pilots in Croatian and Slovene to test the design presented in the previous section. In the pilots so far, annotators were not asked explicitly to rate the words "within the contexts"; while this should have encouraged pure priming effects, minimizing lexical modulation

effects, and the fact that we obtained significant differences is encouraging, we expect that larger and more reliable differences will be obtained if annotators are explicitly told to consider the contexts. Our new pilots therefore use a more explicit question about similarity "in the context of the sentence" in order to promote strong lexical effects.

6. Conclusion

The growing use of context-dependent language models and representations in NLP motivates the need for a dataset against which they can be evaluated, and which can test their ability to reflect human perceptions of context-dependent meaning. CoSimLex will provide such a dataset, and do so across a number of less-resourced languages as well as English. The full dataset will be available for the evaluation stage of Semeval2020 at the beginning of February 2020, and be made public when the competition is over (before the LREC2020 conference).

7. Acknowledgements

This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media). The first author is also supported by the EPSRC and AHRC Centre for Doctoral Training in Media and Arts Technology (EP/L01632X/1).

8. Bibliographical References

Branigan, H. P., Pickering, M. J., and Cleland, A. A. (2000). Syntactic co-ordination in dialogue. *Cognition*, 75(2):B13–B25.

Cruse, D. A. (1986). *Lexical semantics*. Cambridge university press.

Devlin, J., Chang, M.-W., Lee, K., and 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)*, pages 4171–4186.

Evans, V. and Green, M. (2018). *Cognitive linguistics: An introduction*. Routledge.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM Transactions on information systems*, 20(1):116–131.

Gärdenfors, P. (2000). *Conceptual Spaces: The Geometry of Thought*. MIT Press.

Gärdenfors, P. (2014). *The geometry of meaning: Semantics based on conceptual spaces*. MIT Press.

Ginter, F., Hajič, J., Luotolahti, J., Straka, M., and Zeman, D. (2017). CoNLL 2017 shared task - automatically annotated raw texts and word embeddings. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

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

Huang, E. H., Socher, R., Manning, C. D., and Ng, A. Y. (2012). Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 873–882. Association for Computational Linguistics.

Kittask, C. (2019). *Computational Models of Concept Similarity for the Estonian Language*. Bachelor's thesis, University of Tartu.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*.

Mrkšić, N., Vulić, I., Ó Séaghdha, D., Leviant, I., Reichart, R., Gašić, M., Korhonen, A., and 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.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and 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)*, pages 2227–2237. Association for Computational Linguistics.

Pickering, M. J. and Garrod, S. (2004). Toward a mechanistic psychology of dialogue. *Behavioral and brain sciences*, 27(2):169–190.

Pilehvar, M. T. and Camacho-Collados, J. (2019). Wic: the word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1267–1273.

Venekoski, V. and Vankka, J. (2017). Finnish resources for evaluating language model semantics. In *Proceedings of the 21st Nordic Conference on Computational Linguistics, NoDaLiDa, 22-24 May 2017, Gothenburg, Sweden*, number 131, pages 231–236. Linköping University Electronic Press, Linköpings universitet.

Warglien, M. and Gärdenfors, P. (2015). Meaning negotiation. In *Applications of conceptual spaces*, pages 79–94. Springer.

Word1	Word2	SimLex	Context1	Context2	STDev_SL	STDev_C1	STDev_C2
captain	sailor	5.00	5.20	6.44	1.43	1.93	1.77
corporation	business	9.02	9.24	9.51	1.44	0.78	0.69
god	spirit	7.30	5.65	5.30	1.63	2.47	1.90
guilty	ashamed	6.38	7.78	6.14	0.47	1.88	1.73
lawyer	banker	1.88	1.62	2.54	1.18	1.51	2.01
leader	manager	7.27	8.08	7.65	1.43	1.19	1.34
population	people	7.68	6.49	7.73	0.80	2.37	1.92
rabbi	minister	7.62	7.85	8.11	1.35	2.29	1.21
sheep	cattle	4.77	4.37	4.47	0.47	2.36	2.04
task	woman	0.68	0.15	0.15	0.34	0.42	0.40
wealth	prestige	6.07	5.20	6.67	1.55	2.05	1.74

Table 1: Results from the second English pilot including mean ratings and standard deviation for each context, and the original SimLex values for comparison. Rows shown shaded show a significant difference between ratings for Context1 and Context 2 (Mann-Whitney U test $p < 0.05$).

Word1	Word2	Predicted Potential	Context1	Context2	STDev_C1	STDev_C2
bog	duh	Noticeable difference	3.75	2.50	0.96	2.17
čovjek	dijete	Small difference	2.50	4.25	1.76	0.96
ideja	slika	Noticeable difference	3.33	2.00	2.16	0.82
nedavan	nov	Big difference	4.17	3.25	1.47	2.22
područje	regija	Small difference	5.50	5.33	0.58	0.82
presudan	važan	Small difference	5.33	5.00	0.82	0.82
rijeka	dolina	Noticeable difference	0.33	0.75	0.82	0.50
škola	pravo	Noticeable difference	1.75	0.50	2.22	0.84
sunce	nebo	Small difference	1.50	2.50	1.87	1.73
uništiti	izgraditi	Small difference	0.25	0.83	0.50	1.60
velik	težak	Noticeable difference	3.75	1.67	1.71	2.66
znati	vjerovati	Small difference	2.25	2.17	1.71	1.72

Table 2: Results from the Croatian pilot. In addition to the mean score values it shows the *Predicted Potential* for contextual differences, as judged by the single expert annotator. In each case, Context 1 was the context in which the expert annotator expected the words to be perceived as more similar, and Context 2 as less similar (this applies only to order of presentation here, not to the annotators). Rows shown shaded suggest a trend towards significant difference between ratings for Context1 and Context2 (Mann-Whitney U test $p < 0.15$.